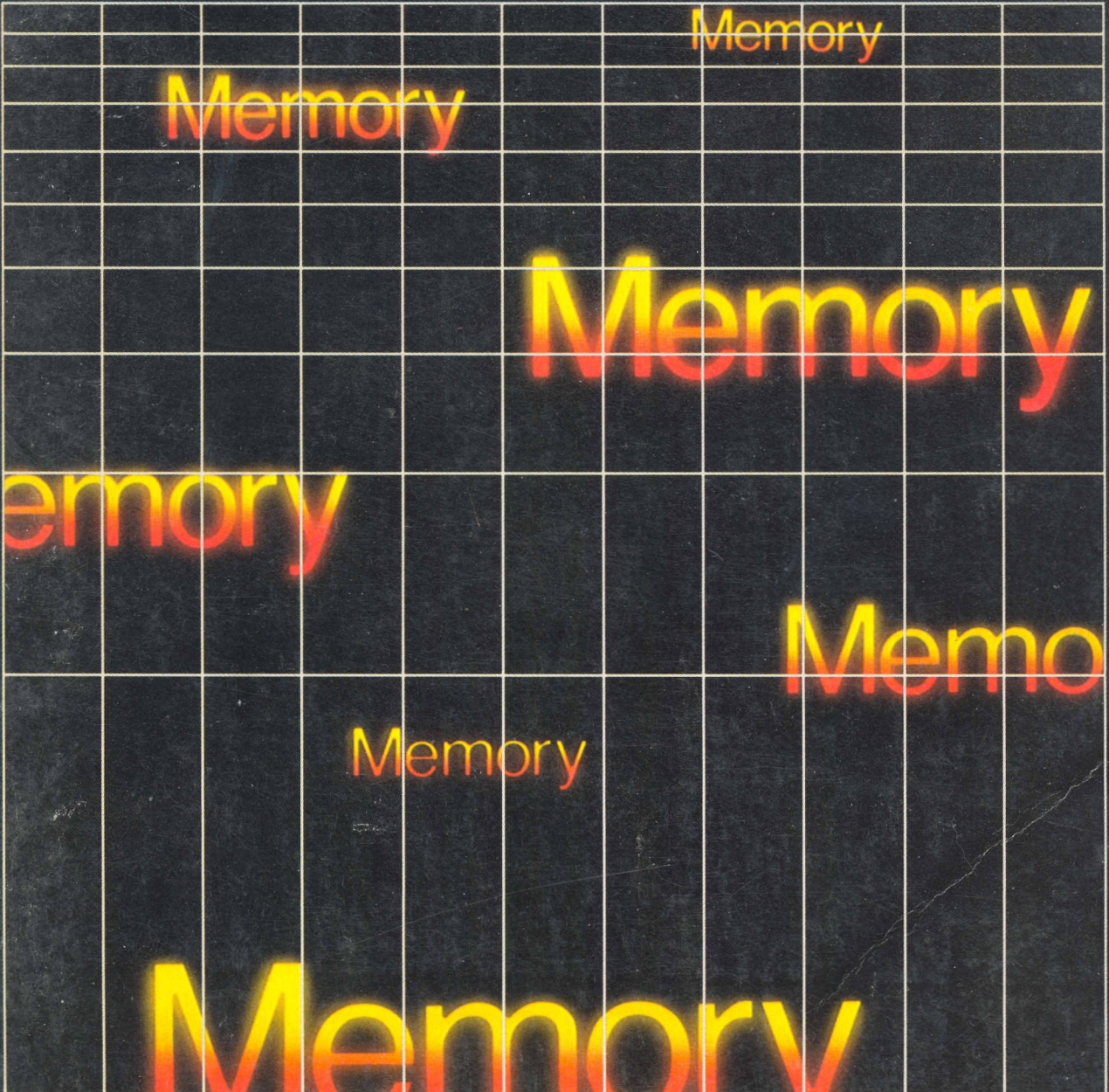


intel

Memory Design Handbook





Bramley 2018

January 1981

January 1981

Intel Corporation makes no warranty for the use of its products and assumes no responsibility for any errors which may appear in this document nor does it make a commitment to update the information contained herein.

Intel software products are copyrighted by and shall remain the property of Intel Corporation. Use, duplication or disclosure is subject to restrictions stated in Intel's software license, or as defined in ASPR 7-104.9 (a) (9). Intel Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in an Intel product. No other circuit patent licenses are implied.

No part of this document may be copied or reproduced in any form or by any means without the prior written consent of Intel Corporation.

The following are trademarks of Intel Corporation and may only be used to identify Intel products:

BXP	Inteleview	MULTIBUS*
CREDIT	Inteltec	MULTIMODULE
i	iSBC	PROMPT
ICE	iSBX	Promware
ICS	Library Manager	RMX
i _m	MCS	UPI
Insite	Megachassis	μScope
Intel	Micromap	

and the combinations of ICE, iCS, iSBC, MCS or RMX and a numerical suffix.

MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.

*MULTIBUS is a patented Intel bus.

Additional copies of this manual or other Intel literature may be obtained from:

Literature Department
Intel Corporation
3065 Bowers Avenue
Santa Clara, CA 95051

Table of Contents

CHAPTER 1	
Introduction	1-1
CHAPTER 2	
Random Access Memories (RAMs)	
AP-74 High Speed Memory System Design Using 2147H	2-1
AP-75 Application of the Intel 2118 16K RAM	2-21
CHAPTER 3	
Programmable Read Only Memories (PROMs)	
RE-1 Bipolar PROM Summary Engineering Report	3-1
Application of the 2716 16K EPROM	3-3
AP-78 Design with EPROMs for Future Flexibility	3-17
AP-93 The Need for Speed—The Future is Now	3-28
RE-2 2732A Reliability Engineering Evaluation Report	3-37
Product Selection Guide	3-42
CHAPTER 4	
Memory System Design Information	
AP-46 Error Detecting and Correcting Codes Part 1	4-1
AP-73 ECC #2 Memory System Reliability with ECC	4-13
RR-26 HMOS II Reliability Report	4-34
A Total System Solution to Magnetic Bubble Memory Applications	4-48
CHAPTER 5	
Article Reprints	
AR-44 Speedy RAM Runs Cool with Power-down Circuitry	5-1
AR-46 HMOS Scales Traditional Devices to Higher Performance Level	5-6
AR-71 Single Supply 16K Dynamic RAM is Ready for Denser Systems	5-12
AR-87 Get Top Memory System Performance at Low Power Levels with MOS RAMs	5-18
AR-111 EPROM Doubles Bit Density without Adding a Pin	5-24
AR-112A Universal Byte Wide Pinout: 2764 Is the Key	5-28
AR-119 16K EE-PROM Relies on Tunneling for Byte Erasable Program Storage	5-38
AR-129 A 35 ns 16K PROM	5-43

Introduction

1

CHAPTER 1

support circuits in system application. It is intended to aid the system designer to gain a thorough understanding of the operation and characteristics of Intel memory components in a system environment.

Random Access Memories (RAMs)

2

[illegible]

CHAPTER 2

AP-74 HIGH SPEED MEMORY DESIGN USING 2147H

INTRODUCTION

The Intel® 2147H is a 4096-word by 1-bit Random Access Memory, fabricated using Intel's reliable HMOS II technology. HMOS II, the second generation HMOS, is Intel's high performance n-channel silicon gate technology, making simple, high speed memory systems a reality. The purpose of this application note is to describe the 2147H operation and discuss design criteria for high speed memory systems.

TECHNOLOGY

When Intel introduced the HMOS 2147, MOS static RAM performance took a quantum leap by combining scaling, internal substrate bias generation, and automatic powerdown. As a result, the 2147 has an access time of 55ns, density of 4096 bits, and power consumption of .99W active and .165W standby.

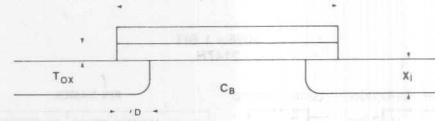
The high performance of the 2147 is further enhanced by the 2147H using HMOS II, a scaled HMOS process increasing the speed at the same power level which involves more than scaling dimensions.

Figure 1 shows the cross section of an HMOS device and lists the parameters of scaling, one of which is high device gain. The slew rate of an amplifier or device is proportional to the gain. Because faster switching speeds occur with high gain, the gain is maximized for high speed. Device gain is inversely proportional to the oxide thickness (T_{OX}) and device length (ℓ), consequently, scaling these dimensions increases the gain.

Another factor which influences performance is unwanted capacitance which appears in two forms - - diffusion and Miller. Diffusion capacitance is directly proportional to the diffusion depth (X_j) into the silicon, thus X_j must be reduced. Miller capacitance, the same phenomenon that occurs in the macro world of discrete devices, is proportional to the overlap length of the gate and the source (ℓ_D). Capacitance on the input shunts the high frequency portion of the input signal so that the device can only respond to low frequencies. Secondly, capacitance from the drain to the gate forms a feedback path creating an integrator or low pass filter which degrades the high frequency performance. This effect is minimized by reducing ℓ_D .

One of the limits on scaling is punch through voltage, which occurs when the field strength is too high, causing current to flow when the device is "turned off". Punch through voltage is a

function of channel length (ℓ) and doping concentration (C_B), thus channel shortening can be compensated by increasing the doping



PERFORMANCE FACTORS

- HIGH DEVICE GAIN
- LOW DIFFUSION CAPACITANCE
- LOW MILLER CAPACITANCE
- LOW BODY EFFECT

$$\begin{aligned} \text{GAIN} &\propto 1/(T_{OX}\ell) \\ C_D &\propto X_j \\ C_m &\propto \ell_D \\ \Delta V_T &\propto \sqrt{C_B} T_{OX} \end{aligned}$$

LIMITS

- PUNCH THROUGH VOLTAGE
- THRESHOLD VOLTAGE

$$\begin{aligned} V_{PT} &\propto C_B \ell^2 \\ V_T &\propto \sqrt{C_B} T_{OX} \end{aligned}$$

RESULT

- DECREASE ℓ , T_{OX} , X_j , ℓ_D
- INCREASE C_B

$$\begin{aligned} \ell &= \text{CHANNEL LENGTH} \\ T_{OX} &= \text{OXIDE THICKNESS} \\ X_j &= \text{DIFFUSION DEPTH} \\ \ell_D &= \text{GATE OVERLAP} \\ C_B &= \text{CONCENTRATION} \end{aligned}$$

Figure 1. HMOS Scaling

concentration. This has the additional advantage of balancing the threshold voltage which was decreased by scaling the oxide thickness for gain.

Comparison

Comparing scaling theory to HMOS II scaling in Table I, note that HMOS II agrees with scaling theory except for the supply voltage. It is left constant at +5V to maintain TTL compatibility. Had the voltage been scaled, the power would have been reduced by $1/K^3$ rather than $1/K$, but the device would not have been TTL compatible.

Table I. Scaling

	Theory	HMOS II
Dimensions	$1/K$	$1/K$
Substrate Doping	K	K
Voltage	$1/K$	1
Device Current	$1/K$	1
Capacitance A/T	$1/K$	$1/K$
Time Delay VC/I	$1/K$	$1/K$
Power Dissipation VI	$1/K^2$	1
Power Delay Product	$1/K^3$	$1/K$

THE DEVICE

The 2147H is TTL compatible, operates from a single +5 volt supply, and is easy to use.

Figure 2 shows the pin configuration and the logic symbol. The 2147H is compatible with the 2147 allowing easy system upgrade. Contained in an industry standard 18-pin dual in-line package the 2147H is organized as 4096 words of 1 bit. To access each of these words, twelve address lines are required. In addition, there are two control signals: CS, which activates the RAM; and WE,

which controls the write function. Separate data input and output are available. Logical operation of the 2147H is shown in the truth table. The output is in the high impedance or three-state mode unless the RAM is being read. Power consumption switches from standby to active under control of \overline{CS} .

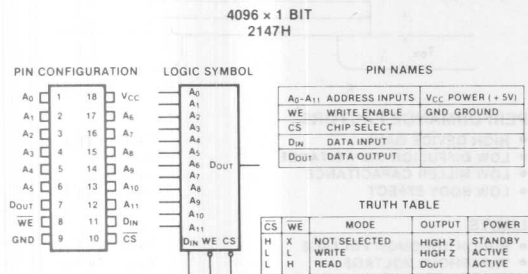


Figure 2. 2147H Logic Diagram

Internal structure of the 2147H is shown in the block diagram of Figure 3. The major portions of the device are: addresses, control (\overline{CS} and \overline{WE}), the memory array and a substrate bias generator, which is not shown.

The memory is organized into a two-dimensional array of 64 rows and 64 columns of memory cells. The lower-order six addresses decode one of 64 to select the row while the upper-order six addresses decode to select one column. The intersection of the selected row and the selected column locate the desired memory cell. Additional logic in the column selection circuit controls the flow of data to the array and as stated in the truth table, \overline{WE} controls the output buffer.

As shown in Figure 4, the first three stages of the address buffer are designed with an additional transistor. In each stage, the lowest transistors are the active devices, the middle transistors are load devices, while the upper transistors, controlled by Φ_1 , are the key to low standby power. Forming an AND function with the active devices, the upper transistors are turned off when the 2147H is not active, minimizing power consumption. Without them, at least one stage of these cascaded amplifiers would always be consuming power.

The signal Φ_1 , and its inverse $\overline{\Phi}_1$, are generated from \overline{CS} . They are part of an innovative design not found in the earlier 2147. Their function is to minimize the effects at short deselection times on the Chip Select access time, t_{ACS} .

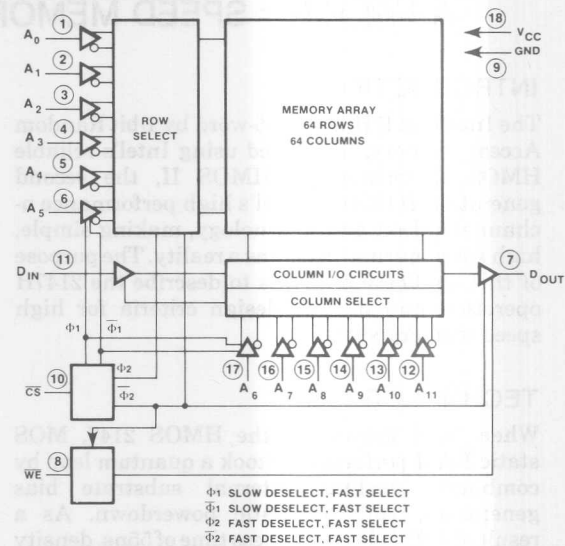


Figure 3. 2147H Block Diagram

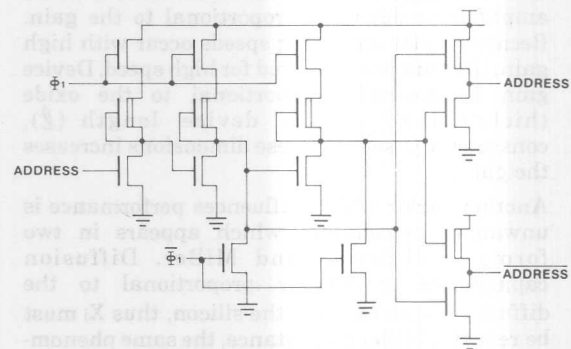


Figure 4. Address Buffer.

For both the 2147 and the 2147H, access is delayed until the address buffers are activated by chip selection. In the standard 2147, priming during deselection compensates for this delay by speeding up the access elsewhere in the circuitry. For short deselection times, however, full compensation does not occur because priming is incomplete. The result is a pushout in t_{ACS} for short deselection times.

In the 2147H, the address buffers are controlled by Φ_1 , which is shaped as shown in Figure 5. Φ_1 activates rapidly for fast select time. However, Φ_1 deactivates slowly, keeping the address buffers active during short deselection times to speed access. As shown in Figure 6, this design innovation keeps t_{ACS} pushout to less than 1 ns.

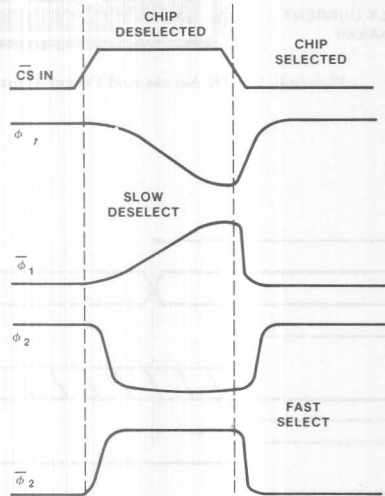


Figure 5. CS Buffer Signals

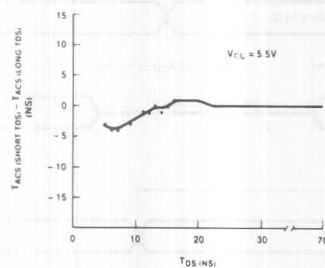


Figure 6. CS Access Vs. Deselect Time

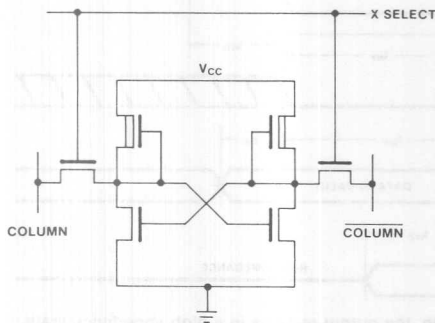


Figure 7. 2147H Memory Cell

Figure 7 shows the standard six-transistor cell. Configured as a bi-stable flip-flop, the memory cell uses two transistors for loads and two for active devices so that the data is stored twice as true and compliment. The two remaining transistors enable data onto the internal I/O bus. Unlike the periphery, the cell is not powered down during deselection time to sustain data indefinitely.

The 2147H has an internal bias generator. Bias voltage allows the use of high resistivity substrate by adjusting the threshold voltages. In addition, it reduces the effect of bulk silicon capacitance. As a result, performance is enhanced. Bias voltage is generated by capacitively coupling the output of a ring oscillator to a charge pump connected to the substrate. Internally generated bias permits the 2147H to operate from a single +5 volt supply, maintaining TTL compatibility.

2147H SUBSTRATE BIAS GENERATOR

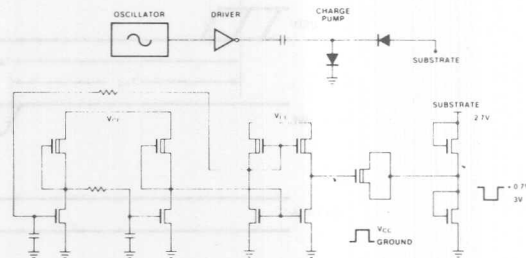


Figure 8. 2147H Substrate Bias Generator

DEVICE OPERATION READ MODE

With power applied and \overline{CS} at greater than 2V, the 2147H is in the standby mode, drawing less than 30mA. Activating \overline{CS} begins access of the cell as defined by the state of the addresses. Data is transferred from the cell to the output buffer. Because the cell is static, the read operation is non-destructive. Device access and current are shown in Figure 9. Maximum access relative to the leading edge of \overline{CS} is 35 ns for a 2147H-1. Without clocks, data is valid as long as address and control are maintained.

WRITE MODE

Data is modified when the write enable \overline{WE} is activated during a cycle. At this time, data present at the input is duplicated in the cell specified by the address. Data is latched into the cell on the trailing edge of \overline{WE} , requiring that setup and hold times relative to this edge be maintained.

Two modes of operation are allowed in a write cycle, as shown in Figure 10. In the first mode, the write cycle is controlled by \overline{WE} , while in the other cycle, the cycle is controlled by \overline{CS} . In a \overline{WE} controlled cycle, \overline{CS} is held active while addresses change and the \overline{WE} signal is pulsed to establish memory cycles. In the \overline{CS} controlled cycle, \overline{WE} is maintained active while addresses again change and \overline{CS} changes state to define cycle length. This flexible operation eases the use and makes the 2147H applicable to a wide variety of system designs.

ADDRESS
INPUT

CHIP SELECT

DATA
OUTPUT

SUPPLY CURRENT
(100 mA/cm)

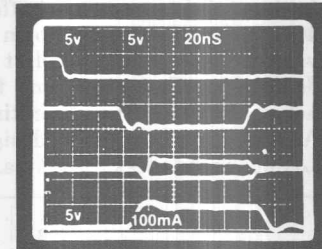
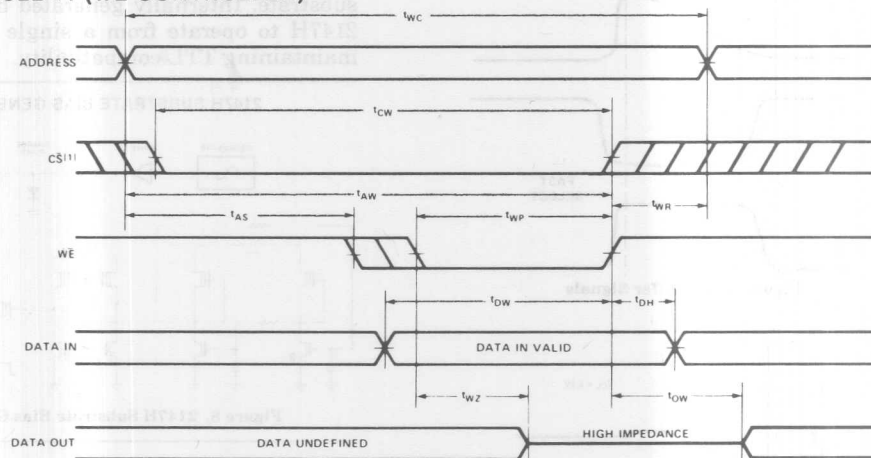


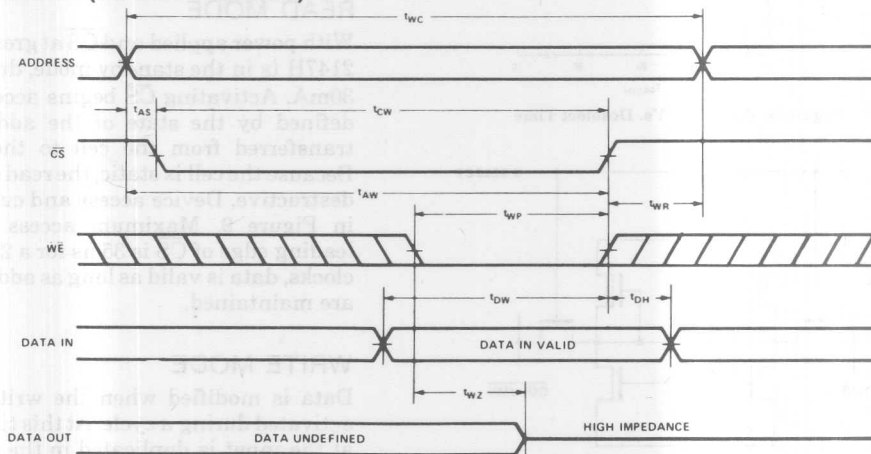
Figure 9. 2147H Access and Power Photo

WAVEFORMS

WRITE CYCLE #1 (\overline{WE} CONTROLLED)



WRITE CYCLE #2 (\overline{CS} CONTROLLED)



Note: 1. If \overline{CS} goes high simultaneously with \overline{WE} high, the output remains in a high impedance state.

Figure 10. Write Cycle Modes of Operation

EFFECT OF POWER DOWN AT THE SYSTEM LEVEL

Power consumed by a memory system is the product of the number of devices, the voltage applied, and the average current:

Equation 1

$$P = NV_{AVE}$$

where:

P = Power

N = Number of devices

V = Voltage applied

I_{AVE} = Average current/device

Without power down, the average current is approximately the operating current. System power increases linearly with the number of devices. With power down, power consumption increases in proportion to the standby current with increasing number of memory devices. Curves in Figure 11 illustrate the difference which results from the majority of devices being in standby with a very small portion of the devices

EFFECT OF POWER DOWN AT THE SYSTEM LEVEL

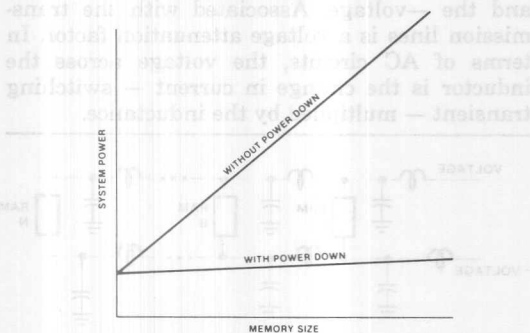


Figure 11. Effect of Power Down at the System

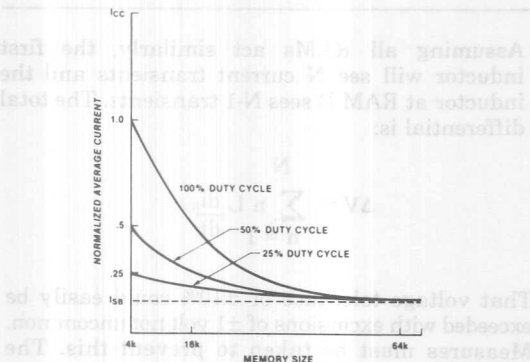


Figure 12. Average Current as a Function of Memory Size

active or being accessed. For a system with power down, the average current of a device in the system is the sum of total active current and the total standby current divided by the number of devices in the system. For an X1 memory such as the 2147H, the number of active devices in most systems will be equal to the number of bits/word, m. Therefore, the number of devices in standby is the difference between N and M. I_{AVE} is expressed mathematically:

Equation 2

$$I_{AVE} = \frac{mI_{ACT} + (N-m)I_{SB}}{N}$$

where:

m = Number of active devices

I_{ACT} = Active current

I_{SB} = Standby current

The graph of Figure 12 shows the relation between average device current and memory size for automatic power down. For large memories the average device current approaches the standby current. Total system power usage, P, is calculated by substituting Equation 2 into Equation 1.

$$P = V[mI_{ACT} + (N-m)I_{SB}]$$

Comparison of power consumption of a system with and without power down illustrates the power savings. Assume a 64K by 18-bit memory constructed with 4KX1 devices. Active current of one device is 180mA and standby current is 30mA. Duty cycle is assumed to be 100% and voltage is 5 volts. The number of devices in the system is:

$$N = \frac{64K \text{ words} \times 18 \text{ bits/word}}{4K \text{ bit/device}}$$

$$N = 288 \text{ devices}$$

WITHOUT POWER DOWN:

$$P_{NPD} = 288 \text{ devices} \times 5 \text{ volts} \times 180 \text{ mA/device}$$

$$P_{NPD} = 259.2 \text{ watts}$$

WITH POWER DOWN:

With power down only 18 devices are active — 18 bits/word — and 270 are in standby.

$$P_{WPD} = 5 \text{ volts} [18 \text{ devices} (180\text{mA/device}) + 270 \text{ devices} (30 \text{ mA/device})]$$

$$P_{WPD} = 56.7 \text{ watts}$$

The system with power down devices uses only 22% of the power required by a non-powerdown memory system.

POWER-ON

When power is applied, two events occur that must be considered: substrate bias start up and TTL instability. Without the bias generator functioning (V_{CC} less than 1.0 volts), the depletion mode transistors within the device draw larger than normal current flow. When the bias generator begins operation (V_{CC} greater than 1.0 volts), the threshold of these transistors is shifted, decreasing the current flow. The effect on the device power-on current is shown in Figure 13.

For V_{CC} values greater than 1.0 v., total device current is a function of both the substrate bias start-up characteristic and TTL stability. During power-on, the TTL circuits are attempting to operate under conditions which violate their specifications; consequently the \overline{CS} signals can be indeterminate. One or several may be low, activating one or more banks of memory. The combined effects of this and the substrate bias start-up characteristic can exceed the power supply rating. The V-I characteristic of a power supply with fold back reduces the supply voltage in this situation, inhibiting circuit operation. In addition, the TTL drivers may not be able to supply the current to keep the \overline{CS} signals deactivated.

One of several design techniques available to eliminate the power-on problem is power supply sequencing. Memory supply voltage and TTL supply voltage are separated, allowing the TTL supply to be activated first. When all the \overline{CS} signals have stabilized at 2.0V or greater, the memory supply is activated. In this mode the memory power-on current follows the curve marked $\overline{CS} = V_{CC}$ in Figure 13.

If power sequencing is not practical, an equally effective method is to connect the \overline{CS} signal to V_{CC} through a 1K Ω resistor. Although this does not guarantee a 2.0V \overline{CS} input; empirical studies indicate that the effect is the same.

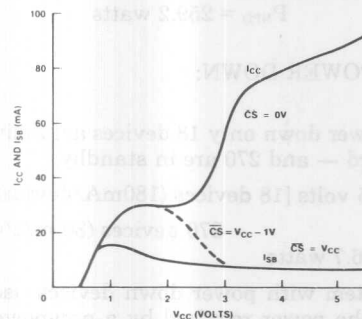


Figure 13. 2147H Power Up Characteristic

ARRAY CHARACTERISTICS

When two or more RAMs are combined, an array is formed. Arrays and their characteristics are controlled by the printed circuit card which is the next most important component after the memory device itself. In addition to physically locating the RAMs, the p.c. board must route power and signals to and from the RAMs.

GRIDDING

A power distribution network must provide required voltage, which from the 2147H data sheet is 5.0 volts $\pm 10\%$ to all the RAMs. A printed circuit trace, being an extremely low DC resistance, should easily route +5v DC to all devices. But as the RAMs are operating, micro circuits within the RAMs are switching micro currents on and off, creating high frequency current transients on the distribution network. Because the transients are high frequency, the network no longer appears as a "pure" low resistance element but as a transmission line. The RAMs and the lumped equivalent circuits of the transmission line are drawn in Figure 14. Each RAM is separated by a small section of transmission line both on the +voltage and the -voltage. Associated with the transmission lines is a voltage attenuation factor. In terms of AC circuits, the voltage across the inductor is the change in current — switching transient — multiplied by the inductance.

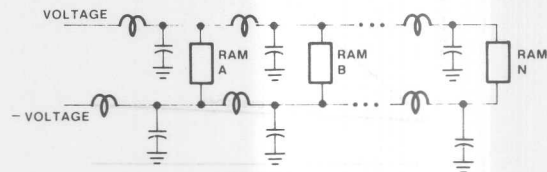


Figure 14. Equivalent Circuit for Distribution

Assuming all RAMs act similarly, the first inductor will see N current transients and the inductor at RAM B sees $N-1$ transients. The total differential is:

$$\Delta V = \sum_{n=1}^N n L \frac{di_n}{dt}$$

That voltage tolerance of $\pm 10\%$ could easily be exceeded with excursions of ± 1 volt not uncommon. Measures must be taken to prevent this. The characteristic impedance of a transmission line is shown in Figure 15A.

Connecting two transmission lines in parallel will halve the characteristic impedance. The result is shown in Figure 15B.

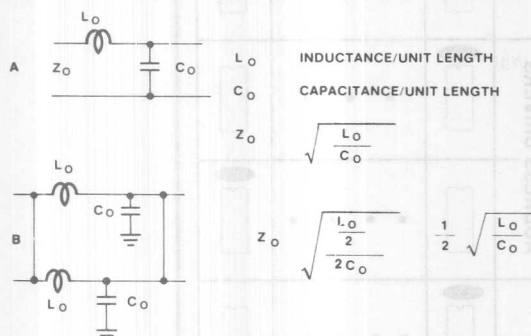


Figure 15. Transmission Line Characteristic Impedance

Paralleling N traces will reduce the impedance to Z_0/N . Extrapolation of this concept to its limit will result in an infinite number of parallel traces such that they are physically touching, forming an extremely wide, low impedance trace, called a plane. Distribution of power (+ voltage) and ground (- voltage) via separate planes provides the best distribution.

P.C. boards with planes are manufactured as multi-layer boards sandwiching the power and ground planes internally. Characteristics of a multilayer board can be cost effectively approximated by gridding the power and ground distribution. Gridding surrounds each device with a ring of power and ground distribution forming many parallel paths with a corresponding reduction of impedance. Gridding is easily accomplished by placing horizontal traces of power (and ground) on one side of the pc board and vertical traces on the other, connected by plated through holes to form a grid.

Viewed from the top of the p.c. board, the gridding as in Figure 16 surrounds each device. Pseudo-gridding techniques such as serpentine or interdigitated distribution, as in Figure 17, are not effective because there are no parallel paths to minimize the impedance.

DECOUPLING

One final aspect of power/ground distribution must be considered - decoupling.

Decoupling provides localized charge to minimize instantaneous voltage changes on the power grid due to current changes. These transient current changes are local and high frequency as devices are selected and deselected. Adequate decoupling

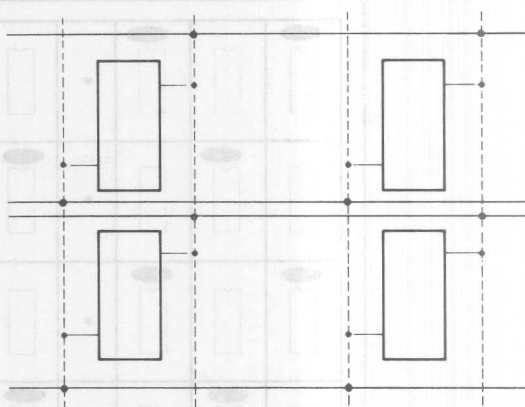


Figure 16. Gridding Plan

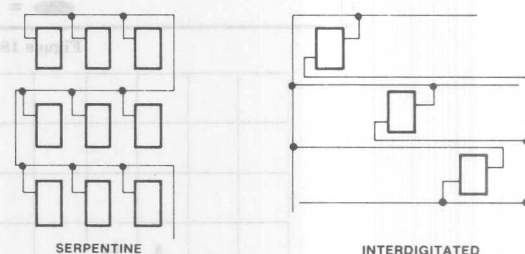


Figure 17. Pseudo-Gridding Techniques

for the 2147H is accomplished by placing a $0.1 \mu\text{f}$ ceramic capacitor at every other device as shown in Figure 18. Bulk decoupling is included on the board to filter low frequency noise in the system power distribution. One tantalum capacitor of 22 to $47 \mu\text{f}$ per 16 devices provides sufficient energy storage. By distributing these capacitors around the board several small currents exist rather than one large current flowing everywhere. Smaller voltage differentials - voltage is proportional to current - are experienced and the voltage remains in the specified operating range. Figure 19 demonstrates the difference with and without gridding.

TERMINATION

Similar reasoning is applied to the a.c. signals: address, control, and data. While they are not gridded or decoupled, they must be kept short and terminated. Similar to the power trace, the signal

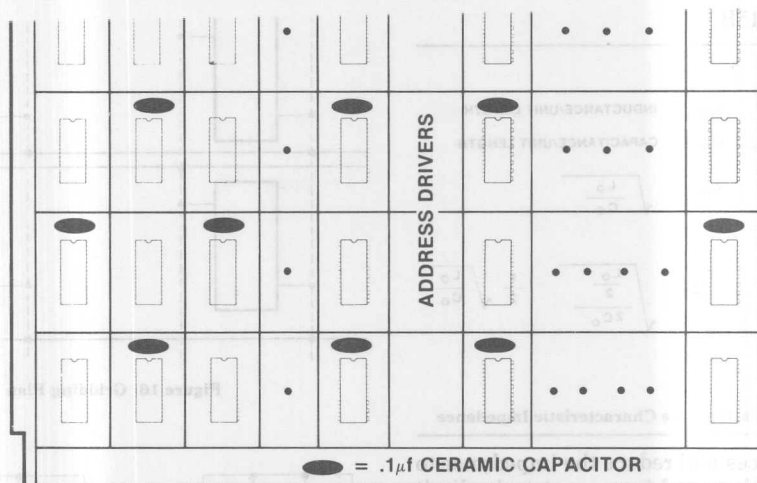
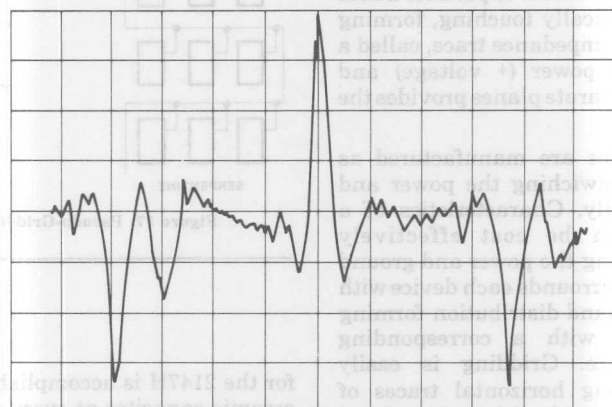
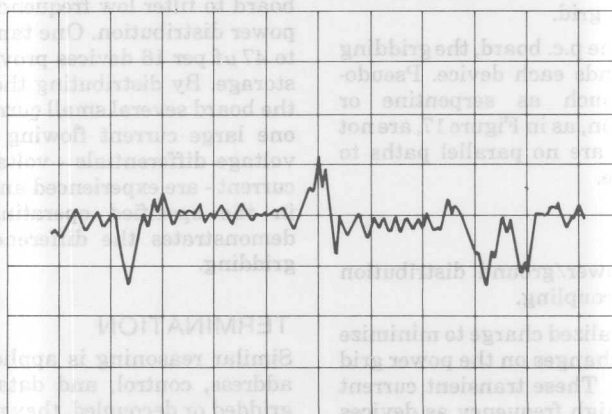


Figure 18. Decoupling



VCC NOISE WITHOUT GRIDDING AND ONE DECOUPLING CAPACITOR PER 4 RAMS



VCC NOISE WITH GRIDDING AND ONE DECOUPLING CAPACITOR PER 2 RAMS

Figure 19. VCC Noise With & Without Gridding

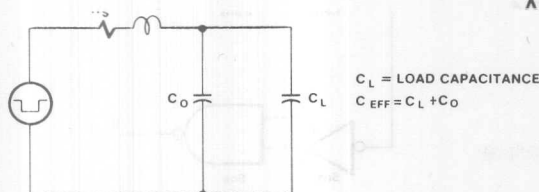


Figure 20. Signal Equivalent Circuit

MOS RAM input is essentially capacitive. Simplifying the capacitance and writing the differential equation.

$$\vartheta = \frac{L di}{dt} + \frac{1}{C} \int i dt$$

The solution of this equation is:

$$i = K_1 e^{-r_1 t} + K_2 e^{-r_2 t}$$

where:

$$r_1 = \frac{R}{2L} + \sqrt{\frac{R^2}{4L^2} - \frac{1}{LC}}$$

$$r_2 = \frac{R}{2L} - \sqrt{\frac{R^2}{4L^2} - \frac{1}{LC}}$$

$K_1 = \text{constant}$

$K_2 = \text{constant}$

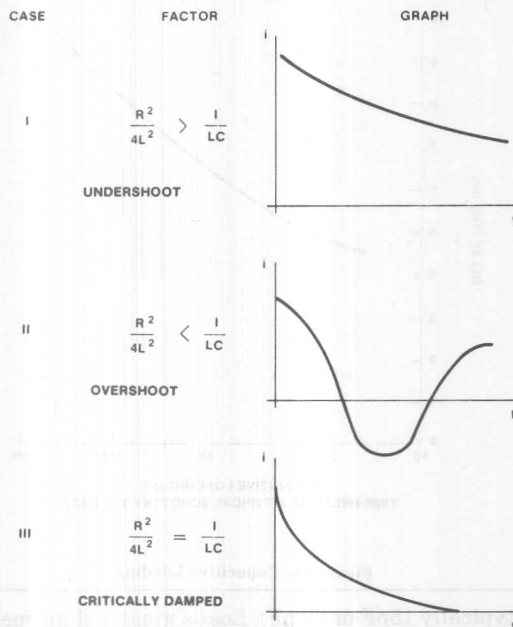


Figure 21. Three Cases of Equation Solution

AD74 current smoothly and clearly changes, while in case II, the current overshoots and rings. If ringing is severe enough, the voltage can cross the threshold voltage of the device as in Figure 22.

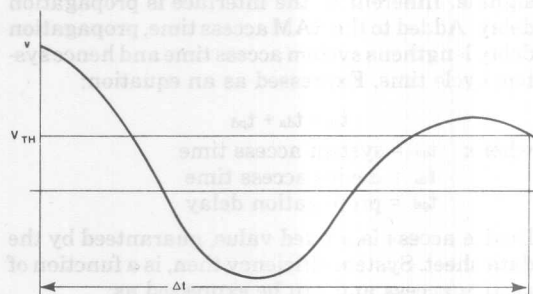


Figure 22. Access Push-Out Due to Ringing

Effective access is stretched out until the wave form settles. System access is the settling time (Δt) plus the specified device access. Case III is the ideal case but in reality a compromise between case I and case II is used because parameters vary in a production environment. Enough series resistance is inserted to prevent ringing but not enough to significantly slow down the access. A series resistance of 33Ω provides this compromise. The exact value is determined empirically but 33Ω is a good first approximation.

SERIES TERMINATION/ PARALLEL TERMINATION

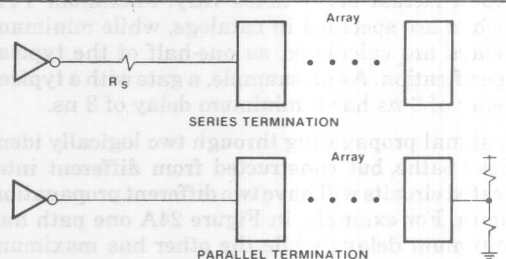


Figure 23. Series and Parallel Termination

Series termination uses one resistor and consumes little power. Current through the resistor creates a voltage differential shifting the levels of input voltage to the devices slightly. This shift is usually insignificant because the 2147H has an extremely high input impedance.

Termination could also be accomplished by a parallel termination as shown in Figure 23.

Parallel termination has the advantage of faster rise and fall times but the disadvantage of higher power consumption and increased board space usage

SYSTEM DELAYS

RAMs are connected to the system through an interface, comprised of address, data and control signals. Inherent in the interface is propagation delay. Added to the RAM access time, propagation delay lengthens system access time and hence system cycle time. Expressed as an equation:

$$t_{sa} = t_{da} + t_{pd}$$

where: t_{sa} = system access time

t_{da} = device access time

t_{pd} = propagation delay

Device access is a fixed value, guaranteed by the data sheet. System efficiency then, is a function of system access and can be expressed as:

$$\text{Eff} = t_{da}/t_{sa}$$

where: Eff = System Efficiency

This can be reduced by substitution for t_{sa} to:

$$\text{Eff} = 1/(1 + t_{pd}/t_{da})$$

System efficiency is maximized when propagation delay is minimized. With sub 100 ns access RAMs, efficiency can be reduced to 40-60% because delay through the signal paths is significant when compared to RAM access. Three factors contribute to the delay: logic delay, capacitive loading, and transit time.

LOGIC DELAY

The delay through a logic element is the time required for the output to switch with respect to the input. Actual delay times vary. Maximum TTL delays are specified in catalogs, while minimum delays are calculated as one-half of the typical specification. As an example, a gate with a typical delay of 6 ns has a minimum delay of 3 ns.

A signal propagating through two logically identical paths but constructed from different integrated circuits will have two different propagation times. For example, in Figure 24A one path has minimum delays while the other has maximum delays. Path A-B has a delay of 3.5 ns while A-B¹ has a delay of 11 ns. The time difference between these two signals is skew, which will be important later in the system design. Figure 24B shows skew values for several TTL devices.

CAPACITIVE LOADING

Delay time is also affected by the capacitive load on the device. Typical delay as a function of capacitive load is shown in Figure 25. TTL data sheets specify the delay for a particular capacitive load

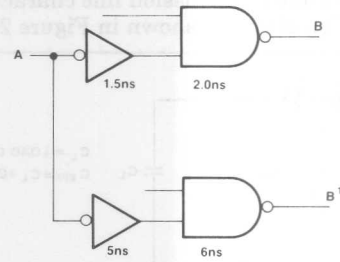


Figure 24A.

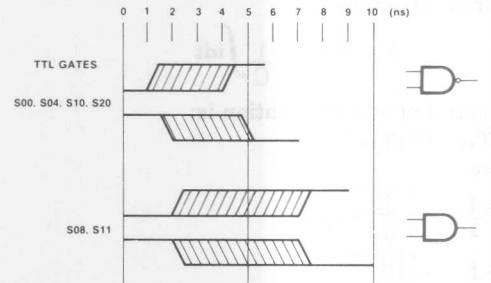


Figure 24B. Skew

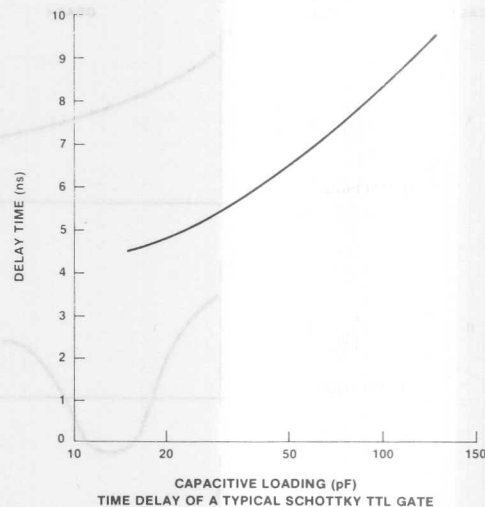


Figure 25. Capacitive Loading

(typically 15pF or 50 pF). Loads greater than specified will slow the device; similarly, loads less than specified will speed up the device.

A value of 0.05 ns/ft is a linear approximation of the function in Figure 25 and is used in the calculations. Loading effect is calculated by subtracting the actual load from the specified load. This difference is multiplied by 0.05 ns/pF and the result algebraically subtracted from the specified delay. As an example, a device has a 4 ns delay driving 50 pF, but the actual load is 25 pF. Then,

$$\begin{array}{r}
 50 \text{ pF specified} \\
 -25 \text{ pF actual} \\
 \hline
 25 \text{ pF difference} \\
 25 \text{ pF} \times 0.05 \text{ ns/pF} = 1.25 \text{ ns} \\
 4 \text{ ns specified} \\
 -1.25 \text{ ns difference} \\
 \hline
 2.75 \text{ ns actual delay}
 \end{array}$$

A device specified at 4 ns while driving 50 pF will have a delay of only 2.75 ns when driving 25 pF. Conversely, the same device driving 75 pF would have a propagation time of 5.25 ns.

TRANSIT TIME

Signal transit time, the time required for the signal to travel down the P.C. trace, must also be considered. As was shown in Figure 19, these traces are transmission lines. Classical transmission line theory can be used to calculate the delay:

$$t_p = \sqrt{LC}$$

where: t_p = Travel Time

L = Inductance/unit length of trace

C = Capacitance/unit length of trace

The capacitance term in the equation is modified to include the sum of the trace capacitance and the device capacitance. This equation approximates in the worst case direction; a signal will never

"see" all the load capacitance simultaneously, it is distributed along the trace at the devices.

Substituting into the equation:

$$tp^1 = \sqrt{L(C + C_L)}$$

where: tp^1 = Modified delay

C_L = Load capacitance

Algebraically:

$$tp^1 = \sqrt{LC(1 + C_L/C)}$$

$$tp^1 = \sqrt{LC} \sqrt{1 + C_L/C}$$

and

$$tp^1 = tp \sqrt{1 + C_L/C}$$

Emperically, tp is 1.8 ns/ft for G-10 epoxy and C is 1.5 pF/in. For a 5-in. trace and a 40 pF load, the delay is calculated to be 4.5 ns. Because this is worst case, an approximated 2 ns/ft can be used. In the following sections, however, the equation will be used. Total delay is the summation of all the delays. Adding the device access, TTL delays and the trace delays result in the system access.

BOARD LAYOUT

The preceding section discussed the effects of trace length and capacitive loading. Proper board layout minimizes these effects.

As shown in Figure 26, address and control lines are split into a right- and left-hand configuration with these signals driving horizontally. This configuration minimizes propagation delay. Splitting the data lines is not necessary, as the data loads are not as great nor are their traces as long as address and control lines. Control and timing fills the remaining space.

Two benefits are derived from this layout. First,

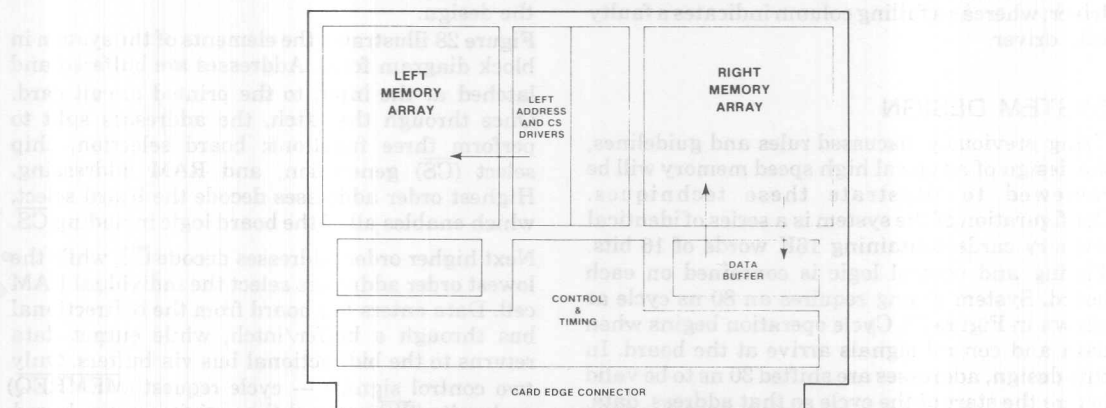


Figure 26. Board Layout

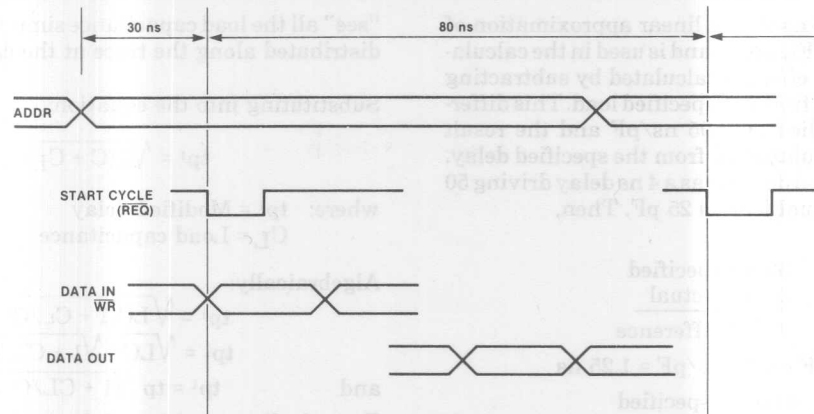


Figure 27. System Timing

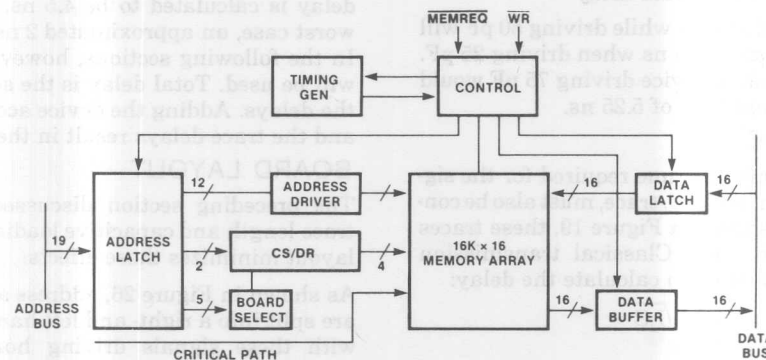


Figure 28. System Block Diagram

the address and control lines are perpendicular to the data lines which minimizes crosstalk. Second, troubleshooting is simplified. A failing row of devices indicates a defective address or control driver; whereas a failing column indicates a faulty data driver.

SYSTEM DESIGN

Using previously discussed rules and guidelines, the design of a typical high speed memory will be reviewed to illustrate these techniques. Configuration of the system is a series of identical memory cards containing 16K words of 16 bits. Timing and control logic is contained on each board. System timing requires an 80 ns cycle as shown in Figure 27. Cycle operation begins when data and control signals arrive at the board. In this design, addresses are shifted 30 ns to be valid before the start of the cycle so that address, data, and control arrive at the memory device at the same time for maximum performance. Data and

control signals are coincident with the start of the cycle. Access is not yet specified because it is affected by device access and the unknown propagation delay. Access will be determined in the design.

Figure 28 illustrates the elements of the system in block diagram form. Addresses are buffered and latched at the input to the printed circuit card. Once through the latch, the addresses split to perform three functions: board selection, chip select (\overline{CS}) generation, and RAM addressing. Highest order addresses decode the board select, which enables all of the board logic including \overline{CS} .

Next higher order addresses decode \overline{CS} , while the lowest order addresses select the individual RAM cell. Data enters the board from the bidirectional bus through a buffer/latch, while output data returns to the bidirectional bus via buffers. Only two control signals — cycle request (\overline{MEMREQ}) and write (\overline{WR}) control the activity on the board.

Figure 29 illustrates the levels of the delay in the

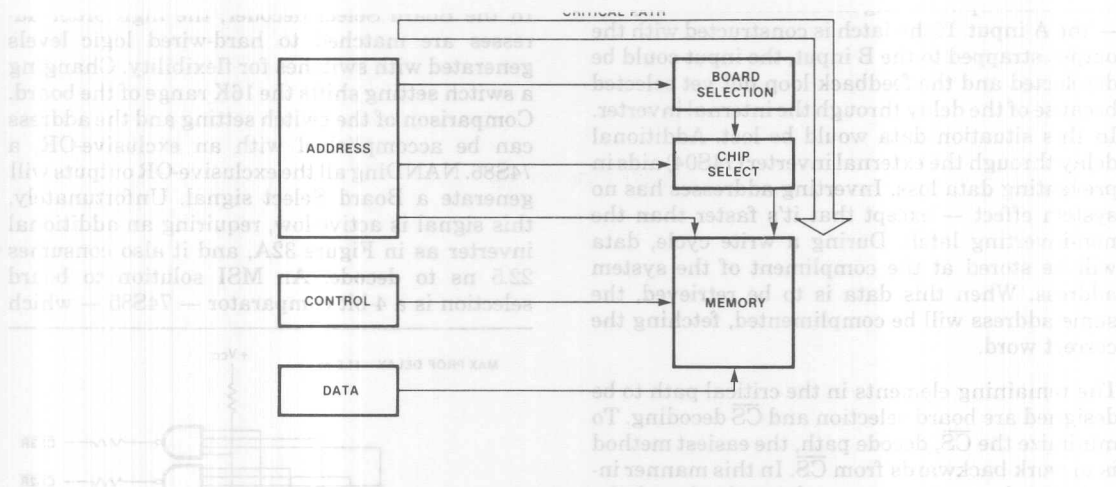


Figure 29. Worst Case Delay Path

system. Data and control have only one level. But examine the address path, it has three levels. Addresses are decoded to activate the logic on the board, select the row of RAM to be accessed and finally locate the specific memory cell. \overline{CS} is in this address path and is crucial for access; without it RAM access cannot begin. But this path has the most levels of decoding with associated propagation delays. Consequently, the address path to \overline{CS} is the critical path and has the greatest effect on system delay and hence must be minimized.

Examination of the system begins with the \overline{CS} portion of the critical path, followed by addresses, data path, and finally timing and control.

CRITICAL PATH

Analysis of the critical path begins with the address latch. The first decision to be made is to the latch type. Latches can be divided into two types: clocked and flow-through. Clocked latches capture the data on the leading or trailing edge of the clock. Associated with the clock is data set-up or hold-time that must be included in the delay time. Accuracy of the clock affects the transit time of the signal because any skew in the clock adds to the delay time. As an example, a typical 74S173 latch has a data set-up time of 5 ns and a maximum propagation delay time from the clock of 17 ns. Total delay time is 22 ns, excluding any clock skew.

Flow-through latches have an enable rather than clock. The enable opens the address window and

allows addresses to pass independent of any clock. Delay time is measured from the signal rather than a clock. The Intel® 3404 is a high speed, 6-bit latch operating in a flow-through mode with 12 ns delay. This is acceptable but a faster latch can be fashioned using a 2-to-1 line multiplexer, either a 74S157 or a 74S158. The slower of the two is the 74S157 with 7.5 ns delay. Although the 74S158 is faster with 6 ns delay, it requires an extra inverter in the feedback path as shown in Figure 30. Between the 74S157 and the 74S158 latches, the trade off is speed against board space and power. Individual designers will choose to optimize their designs.

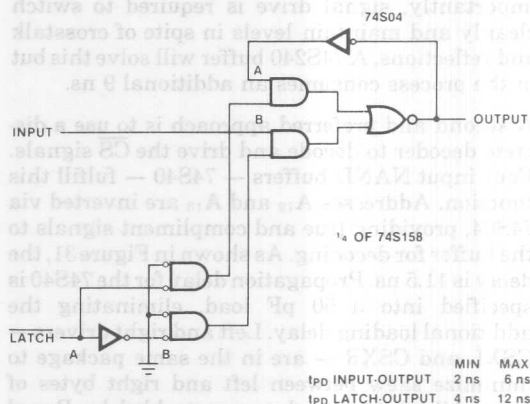


Figure 30. Fast Latch

In either case, care must be exercised in constructing the latch. Output data must be fed back to the input having the shortest internal path — the A input. If the latch is constructed with the output strapped to the B input, the input could be deselected and the feedback loop not yet selected because of the delay through the internal inverter. In this situation data would be lost. Additional delay through the external inverter (74S04) aids in preventing data loss. Inverting addresses has no system effect — except that it's faster than the non-inverting latch. During a write cycle, data will be stored at the compliment of the system address. When this data is to be retrieved, the same address will be complimented, fetching the correct word.

The remaining elements in the critical path to be designed are board selection and \overline{CS} decoding. To minimize the \overline{CS} decode path, the easiest method is to work backwards from \overline{CS} . In this manner input signals to a stage are determined and the output from the preceding stage is defined. This saves inserting an inverter at the cost of 5 ns to generate the proper input to a stage.

Starting with the \overline{CS} driver, the design analyzes several approaches to select the fastest one. With four rows of devices, there are four \overline{CS} signals to be generated. A 2-to-4 line decoder like the 74S138 is a possible solution. It is compact, but has two detriments: long propagation delay and insufficient drive capability. Propagation delay from enable is 11 ns. Enable is driven by board selection which arrives later than the binary inputs. Splitting the RAMs into two 4x8 arrays eases the drive requirement but the demultiplexer must still drive eight devices at 5 pF each — or 40 pF total — which adds 1.75 ns to the delay. More importantly, signal drive is required to switch cleanly and maintain levels in spite of crosstalk and reflections. A 74S240 buffer will solve this but in the process consumes an additional 9 ns.

A second and preferred approach is to use a discrete decoder to decode and drive the \overline{CS} signals. Four input NAND buffers — 74S40 — fulfill this function. Addresses A_{12} and A_{13} are inverted via 74S04, providing true and compliment signals to the buffer for decoding. As shown in Figure 31, the delay is 11.5 ns. Propagation delay for the 74S40 is specified into a 50 pF load, eliminating the additional loading delay. Left and right drivers — CSXL and CSXR — are in the same package to minimize skew between left and right bytes of data. All of the decoders are enabled by Board Select to prevent rows of devices on several boards from being simultaneously active. Board Select is

a true input, defining the output from the Board Select decoder.

In the Board Select decoder, the high order addresses are matched to hard-wired logic levels generated with switches for flexibility. Changing a switch setting shifts the 16K range of the board. Comparison of the switch setting and the address can be accomplished with an exclusive-OR, a 74S86. NANDing all the exclusive-OR outputs will generate a Board Select signal. Unfortunately, this signal is active-low, requiring an additional inverter as in Figure 32A, and it also consumes 22.5 ns to decode. An MSI solution to board selection is a 4-bit comparator — 74S85 — which

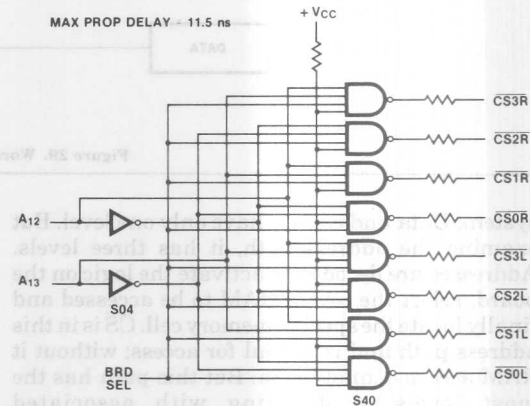
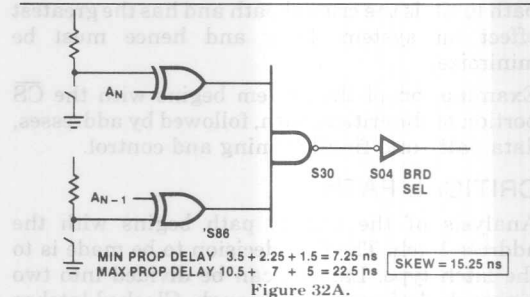
Figure 31. \overline{CS} Decode

Figure 32A.

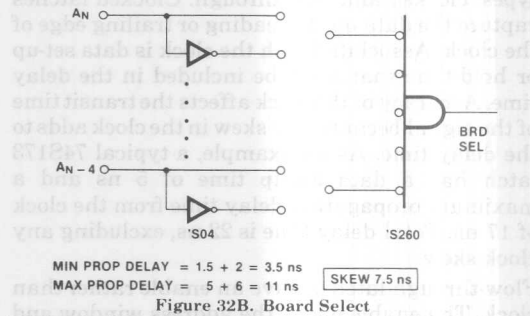


Figure 32B. Board Select

consumes less board area and propagation delay is improved at 16.5 ns.

The best solution is attained by inverting the high order addresses to generate true and complement signals. the appropriate signal is connected into a 74S260, 5-input NOR. With an active-high output, maximum delay is 11 ns as in Figure 32B.

Critical path timing is the sum of the latch, Board Select, and \overline{CS} delay times. In this example, latch delay is 6 ns, Board Select is 11 ns and \overline{CS} decode is 11.5 ns for a total of 28.5 ns. One additional delay — trace delay — must be included for a complete solution. Each 74S40 drives eight MOS inputs having 5 pF/device for a load of 40 pF. Trace capacitance is calculated on 5 in. of trace. At 1.5 pF/in., trace capacitance is 7.5 pF. Trace delay calculated from equation 3 is 1.9 ns.

$$tp^1 = \frac{1.8 \text{ ns} \times 5 \text{ in.}}{\text{ft}} \sqrt{1 + \frac{40 \text{ pF}}{7.5 \text{ pF}}}$$

$$tp^1 = 1.9 \text{ ns}$$

Total worst case maximum critical path delay has been calculated to be 30.4 ns (28.5 ns + 1.9 ns). With the addresses shifted in time by an amount equal to the worst case delay, device and system cycle start are coincident. Start of system access and device access differ only 0.4 ns when the addresses are shifted 30 ns. From the system cycle start, access is stretched by 0.4 ns as shown in Figure 33. Thus, with a 35 ns 2147H-1, data is valid at the output of the device 35.4 ns after the start of the cycle.

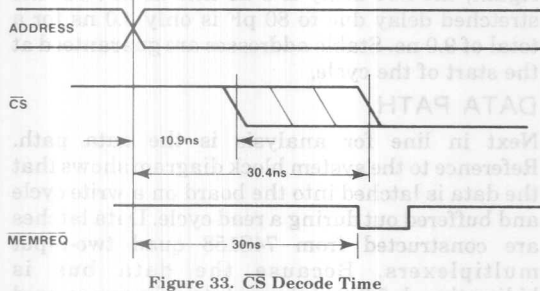


Figure 33. \overline{CS} Decode Time

The minimum delay also must be calculated. With addresses valid prior to the start of the cycle, \overline{CS} decoding can start in the previous cycle. If it occurs too soon, the previous cycle will not be properly completed. Minimum delay time is the sum of the minimum propagation delays plus capacitive loading delay plus trace delay. Capacitive loading delay is less than 0.4 ns and ignored. Minimum delay through the TTL is 9 ns, and added to trace delay results in a total of 10.9 ns.

From address change, the maximum delay in the critical path is 30.4 ns while the minimum is 10.9 ns. The difference between these two times is skew and will be important in later calculations.

ADDRESSES

Lower order addresses (A_0 - A_{11}) arrive at the devices earlier than \overline{CS} because they are not decoded. Consequently, the address drivers do not have a critical speed requirement. Once through the 6 ns latch, addresses have 24 ns to arrive at the devices.

While speed is not the primary prerequisite, drive capability is. Address drivers are located in the center of the board, dividing the array into two sections of 32 devices each. For the moment, assume one driver drives 32 devices as in Figure 34A. Each device is rated at 5 pF/input, resulting in a load of 160 pF. In addition, there are four 5-in. traces — one for each row. twenty inches of trace equates to 30 pF. Total capacitive load is 190 pF. A 74S04 is specified at 5 ns delay into 15 pF. The increased capacitive load is 175 pF, which at 0.05 ns/pF increases the delay by 8.75 ns. Under these conditions the worst cast driver relay is 5 ns plus 8.75 ns, totalling 13.75 ns. It is 10 ns earlier than the 24 ns available.

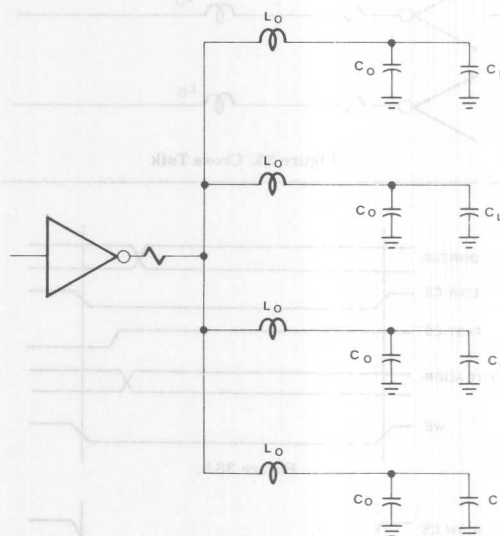


Figure 34A. Address Driver

The first impression is that this is sufficient, but the effect of crosstalk must be considered. For example, as shown in Figure 35, each trace has inductance, and parallel traces take on the

characteristics of transformers. When a signal switches from a one level to a zero level, its driver

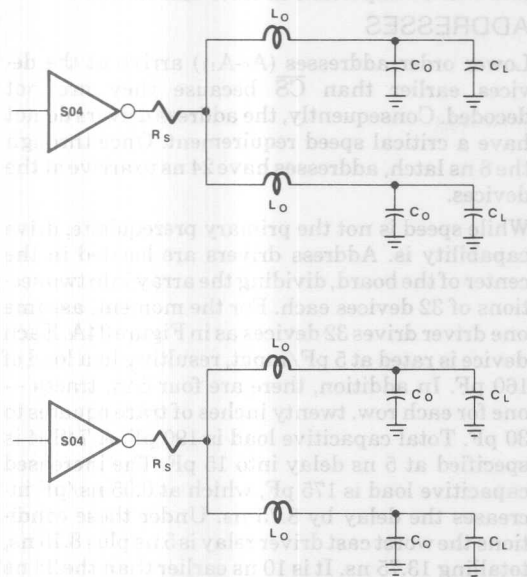


Figure 34B. Address Drivers

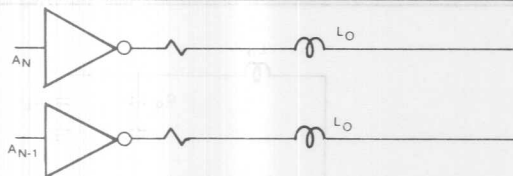


Figure 35. Cross Talk

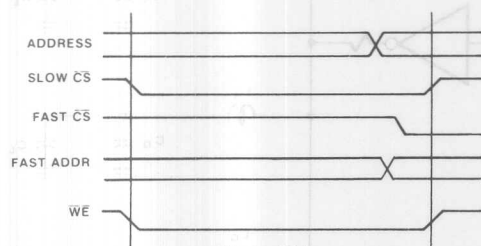


Figure 36A.

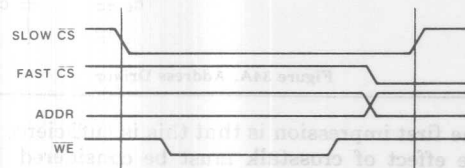


Figure 36B. Race Condition Between Address and WE

can sink 20 mA, inducing a transient in an adjacent trace. If the adjacent signal is switching to a one level, only 400 μ A of a source current from the driver is available. The induced current will generate a negative spike, driving the signal at a one level negative. Additional time of 10 to 15 ns is required to recover and re-establish a stable one level. This may prevent stable address at the start of the cycle. Recall:

$$i = C \frac{dv}{dt} \text{ or } dt = C \frac{dv}{i}$$

where: i = instantaneous current

C = capacitance

$$\frac{dv}{dt} = \text{voltage time rate of change}$$

The term dv/dt can be maximized by increasing i or decreasing C . Current can be doubled by using a driver like a 74S240, but it draws 150mA supply current. In a large system the increased power is a disadvantage because it requires a larger power supply and additional cooling.

A better alternative is to reduce the capacitance, which results in a corresponding increase in dv/dt for quick recovery. Splitting the loads to 16 devices reduces the capacitance and allows a low power driver, like a 74S04, to be used, as in Figure 34B. This has the double effect of decreased propagation delay and providing sharp rise and fall times.

Now, there are only 10 in. of trace or 15 pF load and 16 devices, representing 80 pF for a total of 95 pF. Again, the S04 delay is 5 ns into 15 pF, but the stretched delay due to 80 pF is only 4.0 ns for a total of 9.0 ns. Stable addresses are guaranteed at the start of the cycle.

DATA PATH

Next in line for analysis is the data path. Reference to the system block diagram shows that the data is latched into the board on a write cycle and buffered out during a read cycle. Data latches are constructed from 74S158 quad two-input multiplexers. Because the data bus is bidirectional, 74S240 three-state drivers are used for output buffers.

All that remains to complete the board access computation is the calculation of the output propagation delay. Output delay of the active RAM is caused by the capacitance loading of its own output plus the three idle RAMs, the input capacitance of the 74S240 bus driver and trace capacitance. Output capacitance of the 2147Hs is 6 pF/device for a subtotal of 24 pF; input capacitance of the 74S240 is 3 pF and trace capacitance of a 5-in. trace is 7.5 pF. total load

loading is close enough to the specified loading to eliminate any significant effect on the access calculations. Had there been a difference, the effect would have been included in the calculation. As previously calculated, transit time of the trace is 1.6 ns. Adding this to the 7 ns delay through the 74S240 bus driver results in an 8.6 ns output propagation delay from the RAM output to the bus.

Total access is 35.4 ns plus 8.6 ns output delay for a total access of 44 ns. The efficiency of this system is:

$$\text{Eff} = \frac{35}{44} \text{ or } 80\%$$

TIMING AND CONTROL

Timing and control gating regulates activity on the board to guarantee operation in an orderly fashion. This gating latches addresses, controls the write pulse width and enables the three-state bus drivers. In addition, accurately generated timing compensates for skew effects.

In anticipation of the next cycle, the latch must be opened for the new address. When the current cycle has completed 50 ns, the latches are again opened. The next cycle might not begin 30 ns after the latch is opened because the system may skip one or more memory cycles. Therefore, a signal from the next active cycle must close the latch. In operation, a buffered Memory Request signal latches the addresses.

The write pulse is controlled to guarantee set-up and hold times for data and address and to prevent an overlap of $\overline{\text{CS}}$ and write enable from different cycles. To understand the consequences, consider the following example.

Assume two memory banks, one has a minimum $\overline{\text{CS}}$ and the other has a maximum delay path in $\overline{\text{CS}}$, and both have a minimum address delay. Assume that $\overline{\text{WE}}$ is a level generated from a write command as shown in Figure 36A. The operation under examination is a write cycle into the bank with fast $\overline{\text{CS}}$ followed by a read cycle into the bank with slow $\overline{\text{CS}}$.

Both the write cycle and the read cycle have device specification violations. In the write cycle, the addresses change prior to $\overline{\text{CS}}$ and $\overline{\text{WE}}$ becoming inactive; that new address location may be written into. In the read cycle, the address change is correct but $\overline{\text{WE}}$ is still active and the fast $\overline{\text{CS}}$ begins too soon, performing a non-existent write cycle. Clearly, controlling the width of $\overline{\text{WE}}$ will solve the problems.

Finally, the data output buffers, controlled by timing signals, are enabled only during a read cycle while the board is selected preventing bus contention with two or more boards in the system. More importantly, timing disables the output prior to the start of the next cycle, allowing input data to be stabilized on the bidirectional data bus in preparation for a write cycle.

TIMING GENERATION

Having discussed the philosophy of timing and control, we can now focus on the specifics of address latching, write pulse generation and output-enable timing. To perform these functions timing can be generated from one of three sources: clock and shift register, monostable multivibrator, or delay line.

CLOCKED SHIFT REGISTER

A clocked shift register circuit is shown in Figure 37 consisting of a D-type flip flop and an 8-bit shift register.

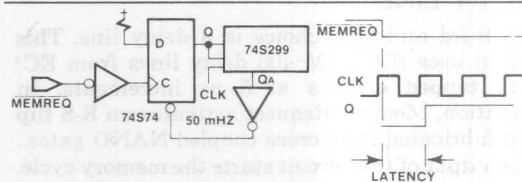


Figure 37. D Flip-Flop and Shift Register

On the leading edge of $\overline{\text{MEMREQ}}$, the Q output of the D flip flop is clocked to a one state, enabling a "one" to be propagated through the shift register. The one is clocked into the first stage of the shift register on the first clock edge after the A and B inputs are "ones". After the clock, the output QA goes true which subsequently clears the D flip flop, clocking zeros into the register to create a pulse one clock period wide.

The accuracy and repeatability depends primarily on the accuracy and stability of the clock. Crystal clocks can be built with +0.005% tolerance and less than a 1% variation due to temperature.

An inherent difficulty is the synchronization of Memory Request and the clock. At times there will be a latency of one clock cycle between Memory Request and the actual start of the cycle when Memory Request becomes active just after the clock edge. Assuming an 80 ns cycle and 20 ns clock, the latency can be 20 ns or 25% of a cycle stretching both access and cycle accordingly. A second difficulty of this circuit is caused by the asynchronous nature of the clock and the Memory Request. The request becomes active just prior to

the clock and the set-up time of the latch is violated, the output QA "hangs" in a quasi-digital state and could double or produce an invalid pulse width; this and the latency hinder effective use in high speed design.

MONOSTABLE MULTIVIBRATOR

The second possible timing generator is a series of monostable multivibrators, using a device such as the AMD Am 26S02 multivibrator. It has a maximum delay from input to output of 20 ns and an approximate minimum of 6 ns. However, with a delay of 20 ns, the monostable multivibrator offers no advantage over the clocked generator. Having a minimum pulse width of 28 ns, the one-shot offers no improvement over the 50 MHz clock, but in fact the performance is worse because it is more temperature and voltage sensitive. The pulse width is dependent on the RC network composed of resistors and capacitors that are temperature sensitive. Consequently, repeatability leaves something to be desired.

DELAY LINE

The third and best choice is a delay line. This design uses STTLDM-406 delay lines from EC² with tapped outputs at 5 ns increments. In operation, Memory Request activates an R-S flip flop fabricated from cross coupled NAND gates. The output of this circuit starts the memory cycle. Consequently, the cycle starts 5 ns after Memory Request compared to 20 ns for the other two timing

generators. The leading edge travels down the delay lines. When the edge reaches the 25 ns tap, the output is inverted and fed back to the R input of the R-S flip flop, shaping the pulse to width to 25 ns. Twenty-five nanoseconds was chosen to match as close as possible the write pulse width. A 25 ns pulse limits the Memory Request signal width to less than 25 ns to insure proper operation. Otherwise, the R-S flip flop will not clear until Memory Request returns to a one level. As the pulse travels down the delay lines, it acquires additional skew of ± 1 ns per delay line package for a total of 6 ns overall. Figure 38 shows several timing pulses and the uncertainty of each edge calculated by worst case timing analysis. The remaining problem is selection of timing edges to operate the device. Now that the timing chain is completely defined, specific details of the address latch, write pulse and output enable can be completed.

ADDRESS LATCH TIMING

An R-S flip flop activated by MEMREQ latches the addresses. A second signal which we will now calculate is used to open the latch. This signal has two boundaries. If the latch opens too late, the access of the cycle will be extended; if it opens too soon, the current cycle will be aborted. Skew through the R-S flip flop is 1.75 ns to 5.5 ns and skew in the latch from enable to output is 4 ns to 12 ns for a total skew of 6 to 17.5 ns. With this skew added to the 30 ns address set-up time, the latch opening signal must be valid at 36 ns best case or

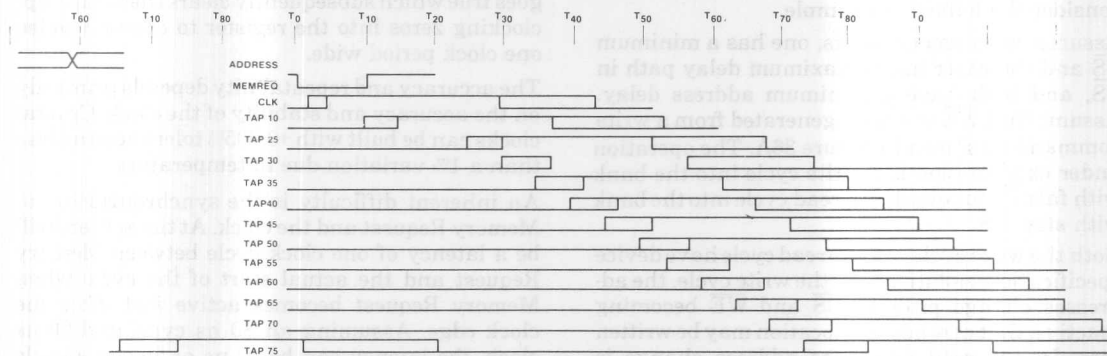


Figure 38. Timing Chain

47.5 ns worst case prior to the start of the memory cycle. Each cycle is 80 ns long, therefore, the latch opening signal must begin 44 ns or 32.5 ns, respectively, in the preceding cycle. From the delay line timing diagram, T35 will satisfy the worst case requirements for opening the latch and T 25 best case. In production, each board is tuned by selecting T25, T30, or T35 to open the latch, guaranteeing it opens between 35 and 30 ns prior to the start of the cycle.

WRITE PULSE TIMING

The next timing to be calculated is the write pulse. Figure 39 shows the three parameters which define the write pulse timing: data set-up time, write pulse width and write recovery time. Data set-up is assured by having data valid through the entire cycle.

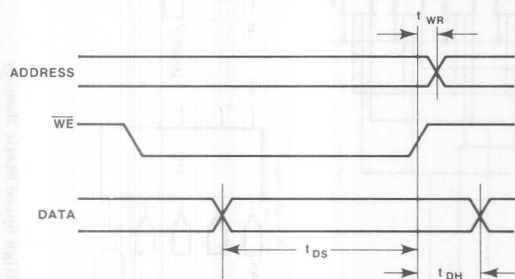


Figure 39. WE Constraints

Placement of \overline{WE} in the cycle is controlled by address change to comply with t_{WR} . From previous calculations the earliest addresses can change is 50 ns, which defines the end of the \overline{WE} signal. Our calculations begin at the device and work back to the timing edge. Eight devices constitute a 40 pF load and a 74S40 is specified for a 50 pF load, reducing delay by 0.5 ns when driving 40 pF. Trace delay and 74S40 delay is 3.5

to 8 ns. Subtracting 8 ns from 50 ns sets the termination of the write timing edge at 42 ns. Using the inversion of T25 will end the write pulse at 43 ns with 7 ns to spare.

Data set-up time is guaranteed because data is valid 6 ns (the worst case delay through the latch) after the start of \overline{MEMREQ} .

OUTPUT ENABLE TIMING

There is a 5.5 ns delay through the address driver providing minimum device cycle of 50 ns. As a result the earliest data can disappear from the bus is at 54 ns because of delay through the output circuit. To select the timing tap for the output enable, the skew of the enable circuit is subtracted from the system access time.

Subtracting the 28 ns skew of the buffer enable circuit from the 44 ns access time of the system shows that the latest the timing edge can occur is 16 ns, which is satisfied by edge T10. The trailing edge, however, ends at 37 ns and with minimum propagation delays the bus would become three-stated at 44 ns, coincident with data becoming valid. ORing T20 with T10 will guarantee the output is valid until 54 ns, minimum. Selecting a timing gap between T35 and T50, depending on the propagation delay in the enable circuit, disables the output at 70 ns, allowing input data to be valid for 10 ns prior to start of cycle. The complete schematic is shown in Figure 40.

SUMMARY

The 2147H is an easy-to-use, high speed RAM. The problems in a memory system design are the result of inherent limitations in interfacing. Largest of these is skew, which the designer must strive to minimize. In this example, skew consumed 45 ns of an 80 ns cycle while device access time was extended by only 10 ns, resulting in an 80% efficiency.

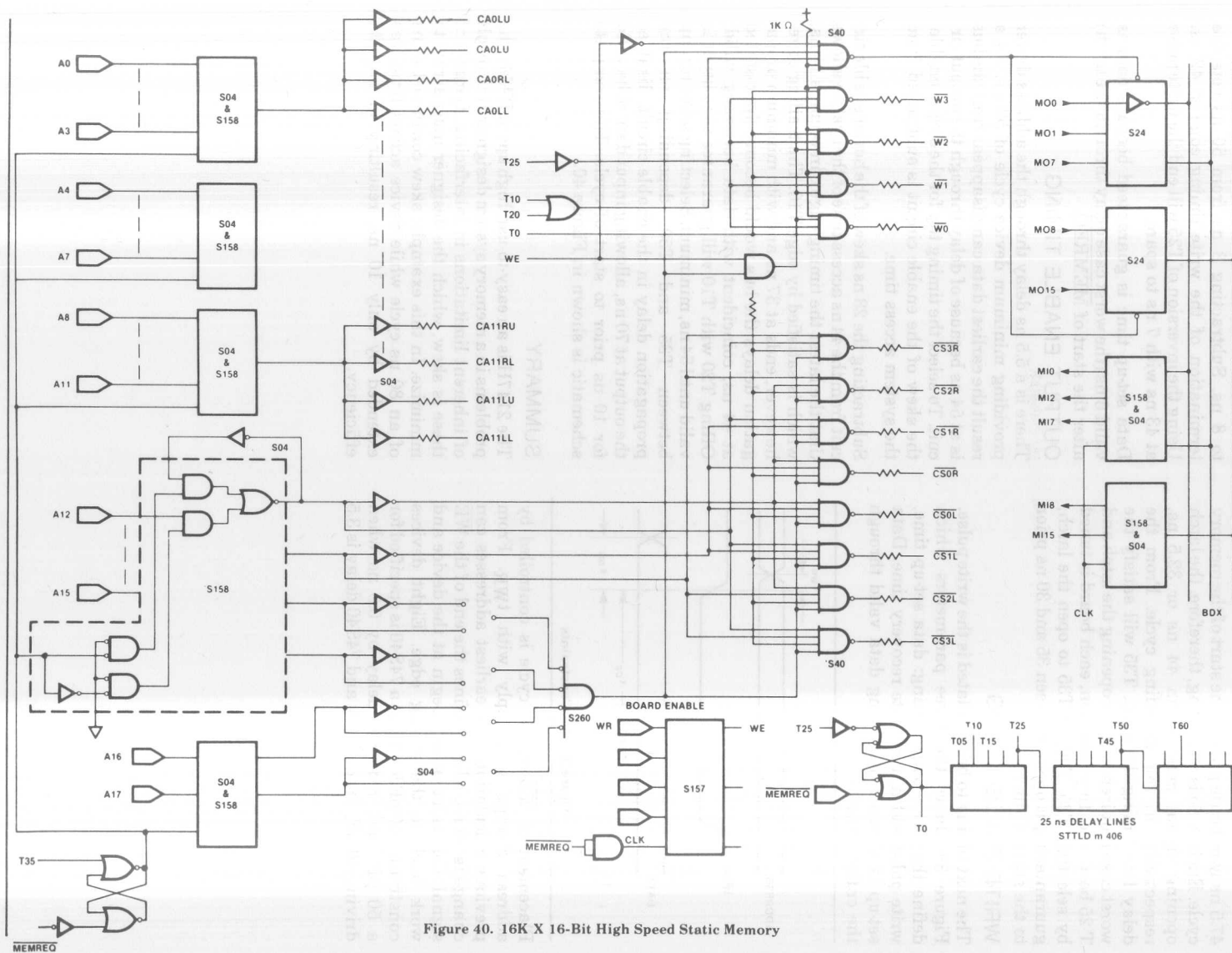


Figure 40. 16K X 16-Bit High Speed Static Memory

AP-75 APPLICATION OF THE INTEL 2118 16K RAM

1. INTRODUCTION

The Intel® 2118 is a high performance, 16,384-word by 1-bit dynamic RAM made possible by Intel's production-proven, advanced n-channel HMOS technology. The Intel 2118 is packaged in the industry standard 16-pin DIP configuration and only requires a single power supply voltage and ground for operation, i.e., V_{DD} (+5V) and V_{SS} (GND). The substrate bias voltage, usually designated V_{BB} , is internally produced by an innovative back-bias generator. This allows additional package pin terminals to be used for other functions such as additional addresses for higher density dynamic RAMs. The single +5V power supply and HMOS reduced geometries result in lower power dissipation and higher performance.

The Intel® 2118 represents a major milestone in design simplicity for the memory system designer. Features such as single +5V operation, input low levels specified at -2V, a wide t_{RCD} timing window, and low power dissipation make dynamic RAMs easier than ever to use.

2. DEVICE DESCRIPTION

Except for significantly reduced power supply requirements, the 2118 is pin compatible with the Intel® 2117, an industry standard 16-pin, 16K RAM. This compatibility provides system upgrade from the 2117 to the 2118 by only having to replace the +12V supply with a +5V supply. The 2118 pin configuration and logic symbols are shown in Figure 1.

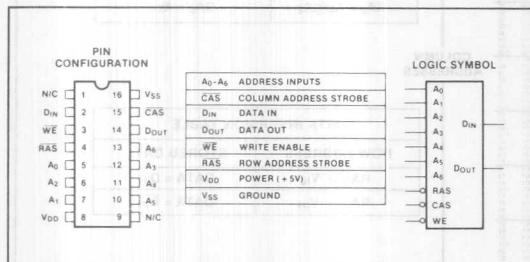


Figure 1. Intel® 2118 Pin Assignments

Fourteen address bits are required to address each of the 16,384 data bits. This is accomplished by multiplexing the address bits onto seven address input pins. The two 7-bit address words are latched into the 2118 by the two TTL level clocks: Row Address Strobe (\overline{RAS}) and Column Address Strobe (\overline{CAS}). Non-critical timing requirements allow the use of the multiplexing technique, while maintaining high performance.

Data is stored in "single transistor" dynamic storage cells. Refreshing is required for data retention and is accomplished automatically by performing a memory cycle (Read, Write or Refresh) at each row address every 2 milliseconds.

3. DEVICE OPERATION

3.1 Addressing

A block diagram of the 2118 is shown in Figure 2. The storage cells are divided into two 8,192-bit memory arrays. Each array is arranged in a 64-row by 128-column matrix. The arrays are connected to a common set of sense amplifiers, column address decoders and I/O lines. When combined, the two arrays create a 128 by 128 matrix which is accessed by row and column addresses present during the \overline{RAS} and \overline{CAS} negative transitions (active cycles). Row Address 0 (RA_0) selects one of the two arrays to be active during any given memory cycle.

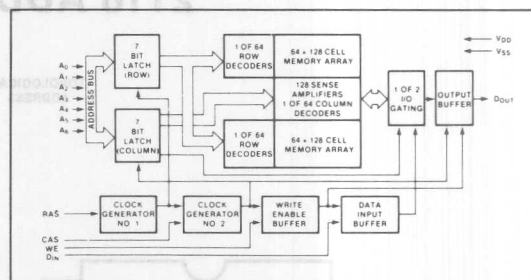


Figure 2. Intel® 2118 Block Diagram

Figure 3 depicts a bit map of the 2118. Table 1 contains the Boolean equations necessary to enable sequential addressing of the 14 required address bits (A_0 - A_{13}). As shown in Table 1, RA_0 is A_6 , RA_1 is A_5 and so forth up to RA_5 and RA_6 which are the "exclusive-OR" of A_1 with A_2 and A_0 with A_2 , respectively. Column addresses are arranged so that the input to CA_6 is the least significant bit of the higher order addresses (A_7) and the remainder of the column addresses as per Table 1. There is no requirement on the user to sequentially address the 2118; the bit map and Boolean equations are shown for information only.

Table 1. Logic Equations for Sequential Addressing of 2118

Sequential Row and Column Decoding	
Row Address	Column Address
$RA_0 = A_6$	$CA_0 = A_{13}$
$RA_1 = A_5$	$CA_1 = A_{12}$
$RA_2 = A_4$	$CA_2 = A_9$
$RA_3 = A_3$	$CA_3 = A_{11}$
$RA_4 = A_2$	$CA_4 = A_{10}$
$RA_5 = A_1 \oplus A_2$	$CA_5 = A_8$
$RA_6 = A_0 \oplus A_2$	$CA_6 = A_7$

3.2 Active Cycles

When row select is activated, 128 cells are sensed simultaneously. A sense amplifier (see Section 3.6) automatically restores the data. When column select is ac-

tivated, Column Address 0 (CA₀) through CA₅ selects one of the 64 column decoders and gates the sensed data from the sense amplifiers onto the two separate differential I/O lines. CA₆ gates one pair into the Data Output Buffer. This data appears as Data Output (D_{OUT}).

Because of independent RAS and $\overline{\text{CAS}}$ circuitry, successive $\overline{\text{CAS}}$ data cycles can be implemented for transferring blocks of data to and from memory at the maximum rate—without reapplying the RAS clock. This procedure is called Page Mode operation and is described in more detail in Section 4.6. If no $\overline{\text{CAS}}$ opera-

tion takes place during the active $\overline{\text{RAS}}$ cycle, a refresh-only operation occurs: RAS-only refresh.

3.3 Back-Bias Generator

The 2118 operates with a single +5V supply. The usual negative power supply voltage is not required because V_{BB} is internally generated by the back-bias generator—a ring oscillator and “charge pumping” circuit. A simplified circuit and an equivalent circuit are shown in Figure 4.

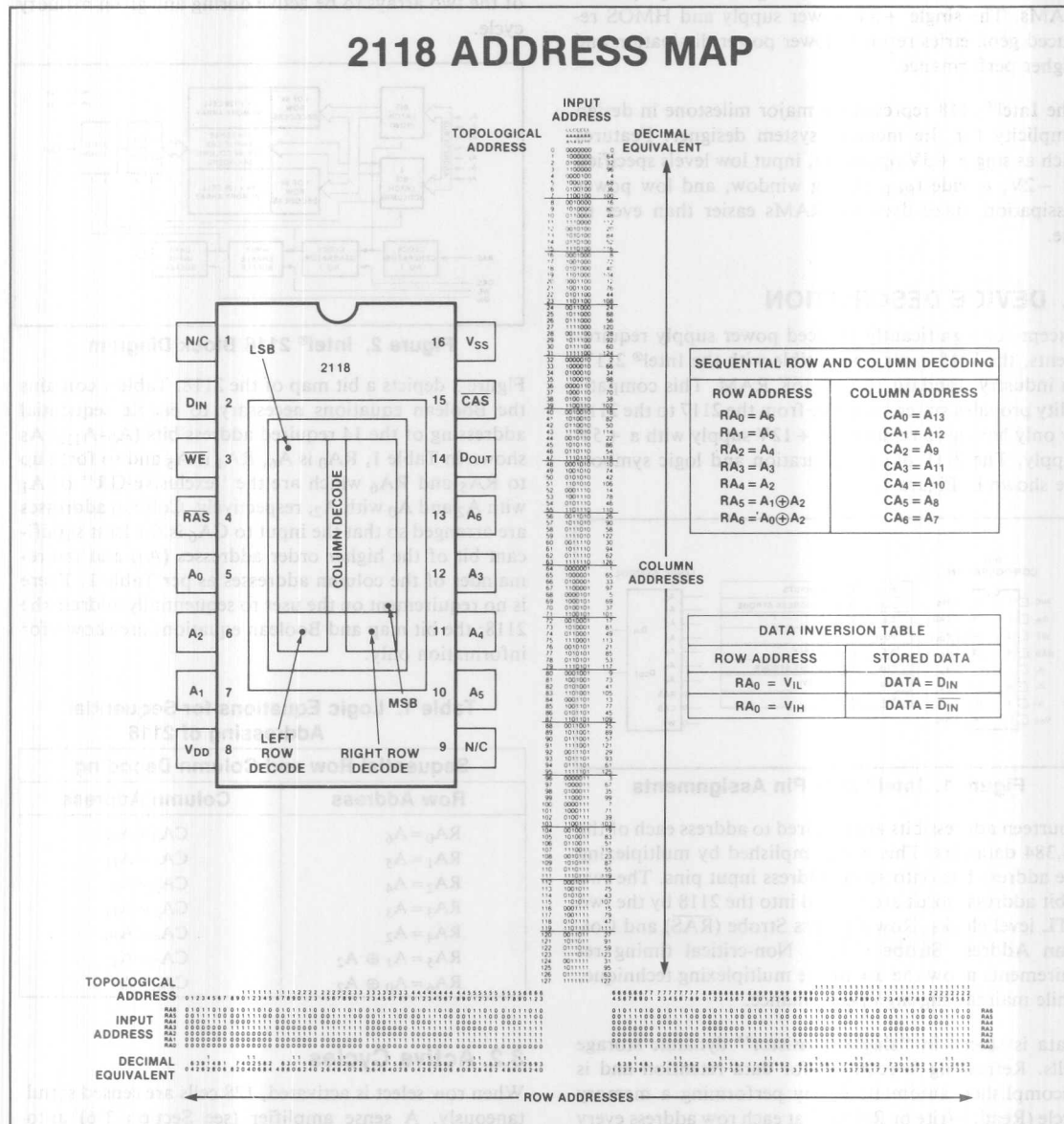


Figure 3. Intel® 2118 Address Map

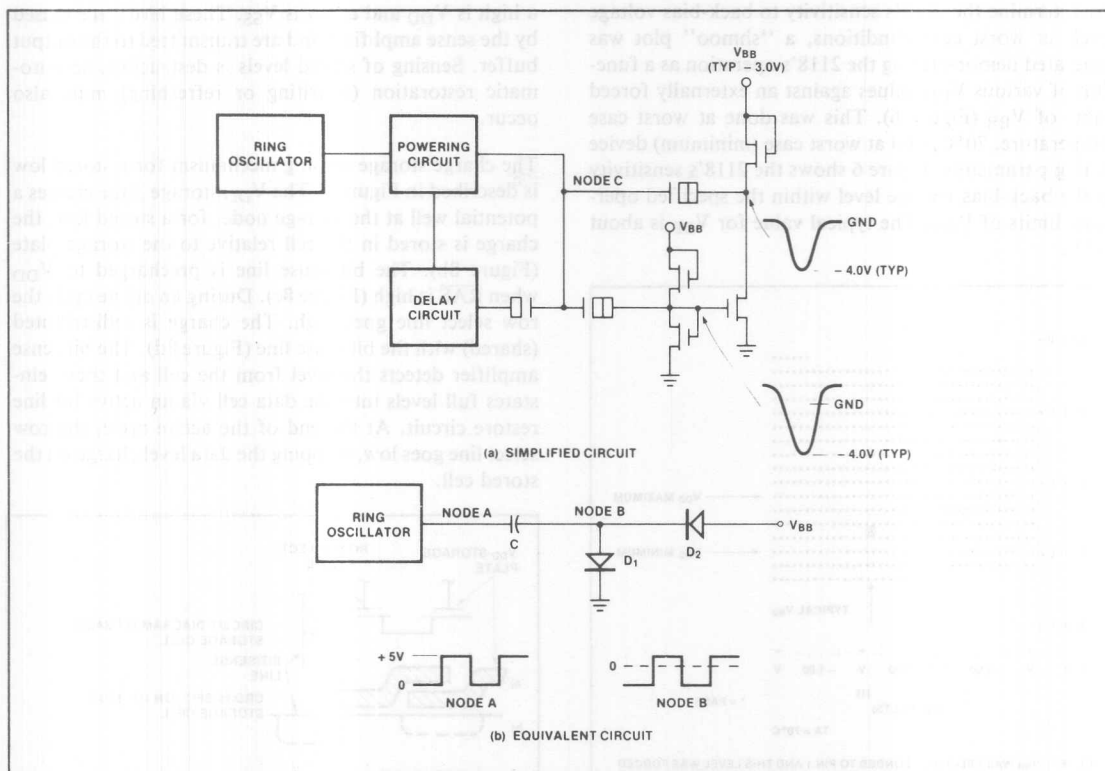


Figure 4. Intel® 2118 Back-Bias Generator

Referring to the equivalent circuit in Figure 4, note that node A is switched between ground and a positive voltage. When node A goes high, capacitor C is charged. Node B will also go positive, but will be clamped by diode D_1 at about +0.6V. The other diode (D_2) is reverse biased and V_{BB} floats. When node A switches to low, node B is capacitively coupled low also. Assuming that V_{BB} is initially at ground potential, this action will cause diode D_1 to reverse bias and diode D_2 to become forward biased at -0.6V. A redistribution of charge between the substrate capacitance, and capacitor C then occurs. The more often this charge redistribution or "pumping" occurs, the more negative the substrate goes. The actual level is dependent on the capacitance ratios of C, the capacitance of the substrate, the voltage level changes at node A and the substrate current.

The 2118 back-bias generator operates in a similar manner to the equivalent circuit but with one major improvement. Node D doesn't go to 0.6V as does node A in the equivalent circuit but is clamped at V_{SS} , allowing for an extra 0.6V of voltage drive. The ring oscillator operates at about 5 MHz and drives the power circuit and delay circuit. The power circuit switches node C between 0 and 5V. Since the delay circuit clamps node D at V_{SS} , a full 5V charge exists on the capacitor (C_2). Node D switches negative and the same charge pumping from the substrate occurs.

The bias doesn't remain indefinitely because the reverse leakage current of the junctions and the impact ionization currents created by the short channel devices cause it to go positive. These currents average about 1 microampere, whereas the generator is capable of supplying currents in excess of 5 microamperes. Thus, the generator can maintain an adequate substrate bias level (Figure 5).

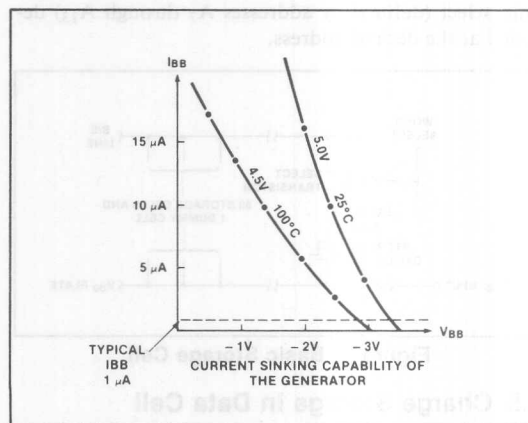


Figure 5. Intel® 2118 Back-Bias Generator Characteristics

generated demonstrating the 2118's operation as a function of various V_{DD} values against an externally forced value of V_{BB} (Figure 6). This was done at worst case temperature, 70°C, and at worst case (minimum) device timing parameters. Figure 6 shows the 2118's sensitivity to the back-bias voltage level within the specified operating limits of V_{DD} . The typical value for V_{BB} is about -3V.

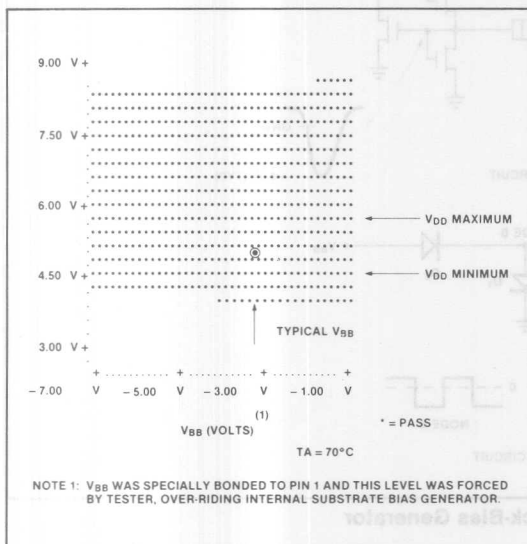


Figure 6. 2118 V_{DD} vs V_{BB} Shmoo Plot

3.4 Storage Cell

The basic storage cell is shown in Figure 7. Data is stored in "single transistor" dynamic storage cells. Each cell consists of a single transistor and a "storage" capacitor. A cell is accessed by the coincidence of a row select (defined by address bits A_0 through A_6) and column select (defined by addresses A_7 through A_{13}) decoded at the desired address.

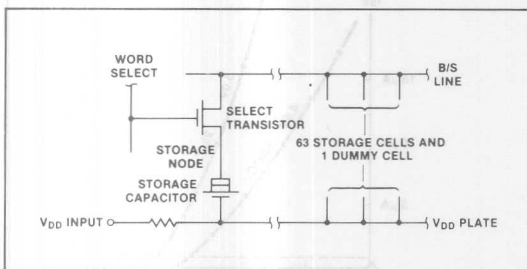


Figure 7. Basic Storage Cell

3.5 Charge Storage in Data Cell

Data is stored in the 2118 storage cells as one of the two discrete voltage levels across the "storage" capacitor—

buffer. Sensing of stored levels is destructive, so automatic restoration (rewriting or refreshing) must also occur.

The charge storage sensing mechanism for a stored low is described in Figure 8. The V_{DD} storage plate creates a potential well at the storage node. for a stored low, the charge is stored in the cell relative to the storage plate (Figure 8b). The bit sense line is precharged to V_{DD} when \overline{RAS} is high (Figure 8c). During an active cycle the row select line goes high. The charge is redistributed (shared) with the bit sense line (Figure 8d). The bit sense amplifier detects the level from the cell and then reinstates full levels into the data cell via an active bit line restore circuit. At the end of the active cycle, the row select line goes low, trapping the data level charge on the stored cell.

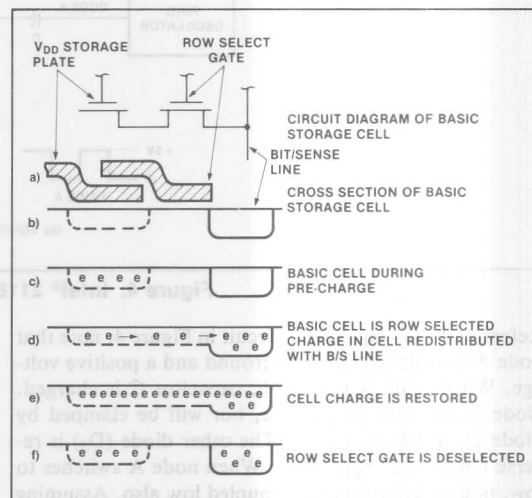


Figure 8. Basic Sensing Mechanism of One Transistor Cell

3.6 Data Sensing

The 2118 sense amplifier compares a stored level to a reference level in a special, non-addressable storage cell. The charge stored in the reference cell (dummy) is less than the minimum allowable stored high level and greater than the maximum allowable stored low level.

Figure 9 depicts a simplified schematic of the Bit/Sense Amplifier (B/S Amp). The B/S Amp consists of a pair of cross-coupled transistors (Q_1 and Q_2), two isolation transistors (Q_3 and Q_4) and a common node (node A) which goes negative and activates the B/S Amp. Bit/sense lines on each side of the B/S Amp interface the storage cells. Each line contains 64 data cells and one dummy cell. Thus each B/S Amp is associated with 128 data cells and two dummy cells. The B/S Lines are precharged by pull-up transistors Q_5 and Q_6 .

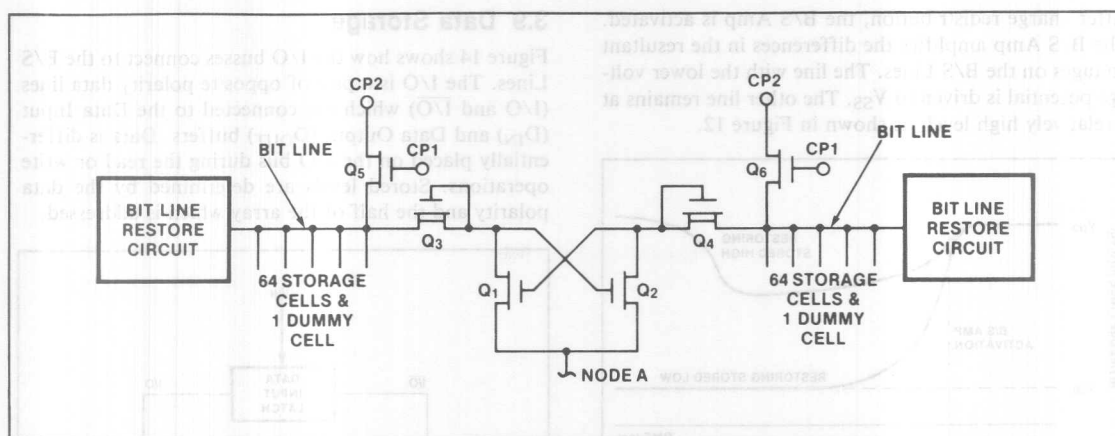


Figure 9. Intel® 2118 Simplified Sense Amplifier Circuit

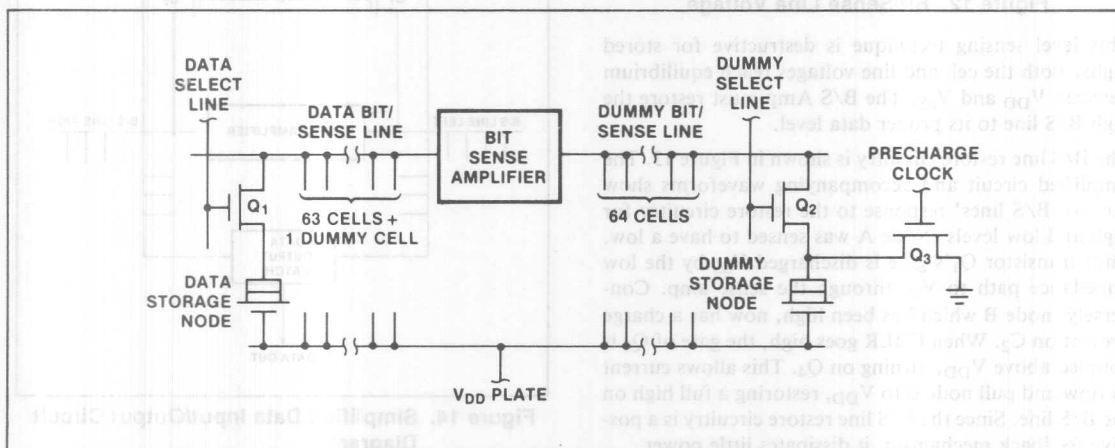


Figure 10. B/S Amplifier and Associated Cells

3.7 Precharge

A precharged period is required after any active cycle to ready the memory device for the next cycle. This occurs as \overline{RAS} goes high. The B/S lines are precharged to V_{DD} , while the dummy cell is precharged to V_{SS} . During precharge, the row select and dummy select lines are at V_{SS} , isolating the cells from the B/S lines. When \overline{RAS} goes low, the precharge clock goes low, ending the precharge period.

3.8 Data Sensing Operation

The row select and dummy select gating are arranged so the selected data and dummy cells are on opposite sides of the B/S Amp (Figure 10). The row select and dummy select lines go high simultaneously resulting in concurrent charge redistribution to the B/S Lines. The relationship between the word select lines and the effect of concurrent charge redistribution on the B/S Lines is shown in Figure 11. An approximate 200 mV differential results from this charge redistribution.

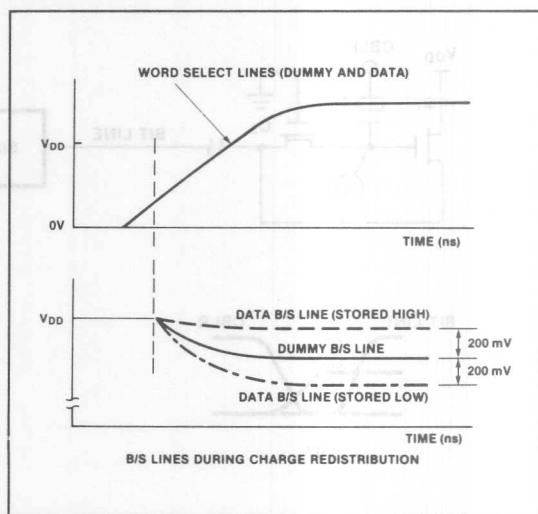


Figure 11. Typical Voltage Waveforms for 2118

After charge redistribution, the B/S Amp is activated. The B/S Amp amplifies the differences in the resultant voltages on the B/S Lines. The line with the lower voltage potential is driven to V_{SS} . The other line remains at a relatively high level, as shown in Figure 12.

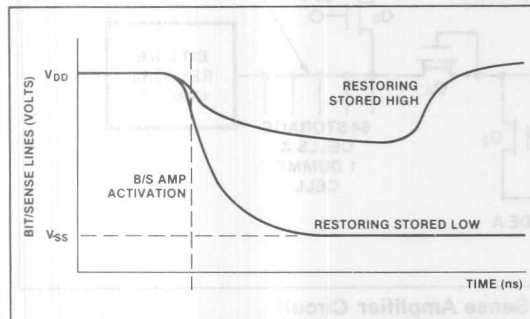


Figure 12. Bit/Sense Line Voltage

This level sensing technique is destructive for stored highs. Both the cell and line voltages reach equilibrium between V_{DD} and V_{SS} . The B/S Amp must restore the high B/S line to its proper data level.

The B/S line restore circuitry is shown in Figure 13. The simplified circuit and accompanying waveforms show the two B/S lines' response to the restore circuitry for high and low levels. Node A was sensed to have a low. Thus transistor Q_1 's gate is discharged V_{SS} by the low impedance path to V_{SS} through the sense amp. Conversely, node B which has been high, now has a charge present on C_2 . When CBLR goes high, the gate of Q_4 is coupled above V_{DD} , turning on Q_4 . This allows current to flow and pull node B to V_{DD} , restoring a full high on the B/S line. Since the B/S line restore circuitry is a positive feedback mechanism, it dissipates little power.

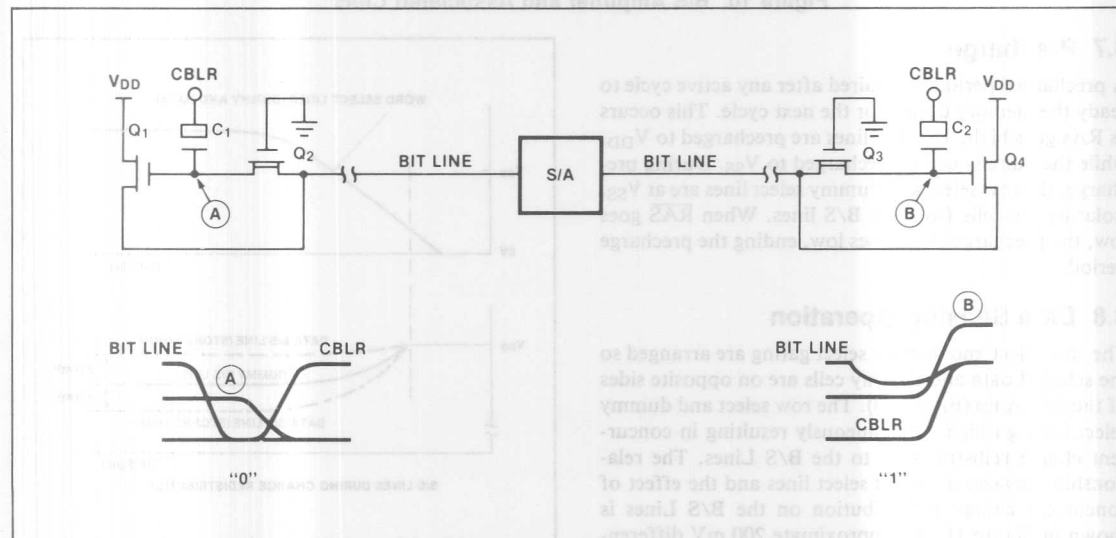


Figure 13. Intel® 2118 Bit Line Restore Circuit

3.9 Data Storage

Figure 14 shows how the I/O busses connect to the B/S Lines. The I/O is a pair of opposite polarity data lines (I/O and $\bar{I/O}$) which are connected to the Data Input (D_{IN}) and Data Output (D_{OUT}) buffers. Data is differentially placed on the I/O bus during the read or write operations. Stored levels are determined by the data polarity and the half of the array which is addressed.

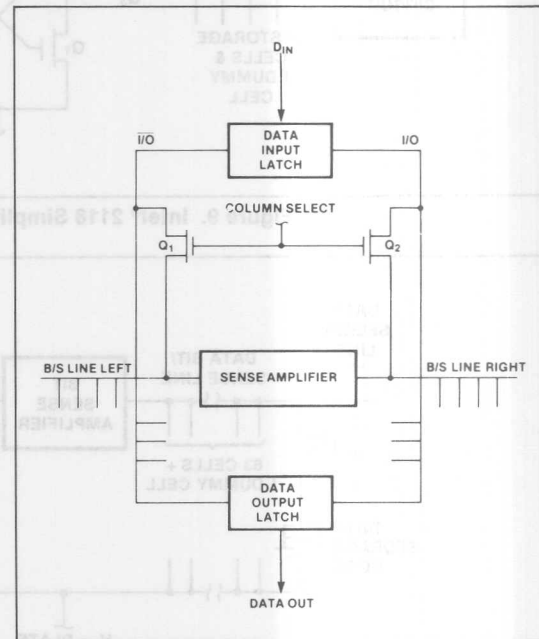


Figure 14. Simplified Data Input/Output Circuit Diagram

Table 2 shows the relationship between D_{IN} , RA_0 and the stored data. For $RA_0 = V_{IL}$, stored data is the same polarity as D_{IN} . For $RA_0 = V_{IH}$, the stored data is opposite. Although the data levels stored are invisible to the user, it is sometimes desirable to know the stored level for testing considerations.

Table 2. Data Storage Level Map of Intel® 2118

RA_0	Data Written (D_{IN})	Stored Level
V_{IL}	V_{IL}	Low
V_{IL}	V_{IH}	High
V_{IH}	V_{IL}	High
V_{IH}	V_{IH}	Low

3.10 Addresses Latches

The 7-bit row and column address words are latched into internal address buffer registers by \overline{RAS} and \overline{CAS} . \overline{RAS} strobes in the seven low-order addresses (A_0 – A_6) both to select the appropriate data select and dummy select lines and to begin the timing which enables the B/S Amps. \overline{CAS} strobes in the seven high-order addresses (A_7 – A_{13}) to select one of the column decoders which enables I/O operation.

Figure 15 shows a simplified input buffer circuit. The address input level is transferred via Q_1 onto the gate of Q_3 during precharge (ϕ_1 high). Similarly, an internally

generated reference voltage (V_{REF}) goes to Q_4 's gate. When ϕ_1 goes low, this charge is held at both gates via C_1 and C_2 . V_{REF} is an internally generated level about halfway between an input high (2.4V) and an input low (0.8V).

This type of address buffer is unique in its use of depletion mode transistors such as Q_3 , Q_4 , Q_5 and Q_6 . Depletion mode transistors are normally on. Thus, in contrast to enhancement mode devices which have a threshold of about 0.8V to 1.0V, there is no threshold sensitivity in the discrimination point at the input. Being always on, the devices provide excellent differential current sources for the cross-coupled latch used to sense the input states.

The combination of substrate bias and high speed input buffers allows input low levels of $-2V$ and extremely short address hold (t_{RAH}) times. This is an important specification when designing high speed switching circuitry driving highly capacitive address busses. Allowing negative overshoots on the address lines means minimum termination of address drivers and increased system performance. This is because a terminated signal (Figure 16) has a slower transition and hence a delay in access time. It is important to note the two advantages to this type of address buffer; first, increased operating speed, and second, a more generous timing window in the multiplexing of the address words.

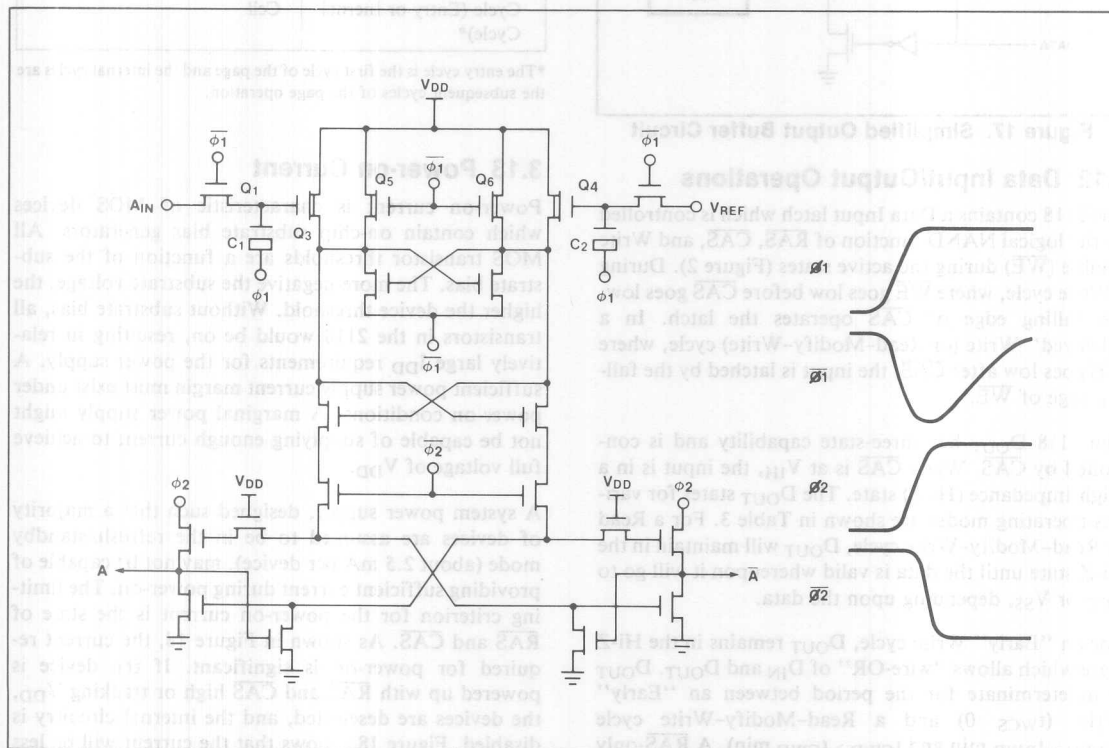


Figure 15. Intel® 2118 Simplified Address Buffer Circuit

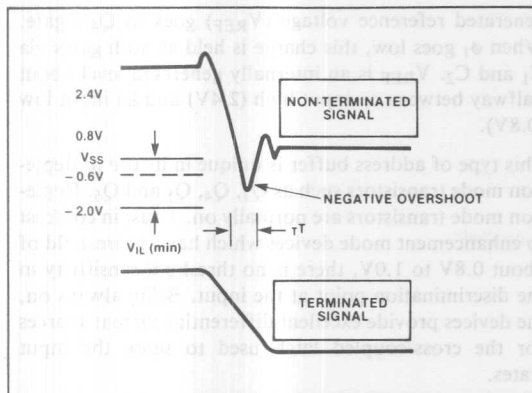


Figure 16. TTL Switching Overshoot Characteristics

3.11 Data Output Buffer

As shown in Figure 17, the buffer has a push-pull transistor configuration in which no dc power is dissipated when active.

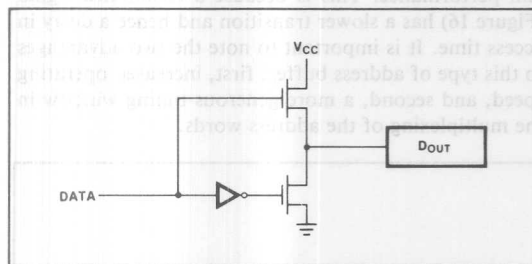


Figure 17. Simplified Output Buffer Circuit

3.12 Data Input/Output Operations

The 2118 contains a Data Input latch which is controlled by the logical NAND function of \overline{RAS} , \overline{CAS} , and Write Enable (\overline{WE}) during the active states (Figure 2). During a Write cycle, where \overline{WE} goes low before \overline{CAS} goes low, the falling edge of \overline{CAS} operates the latch. In a "delayed" Write (or Read-Modify-Write) cycle, where \overline{WE} goes low after \overline{CAS} , the input is latched by the falling edge of \overline{WE} .

The 2118 D_{OUT} has three-state capability and is controlled by \overline{CAS} . When \overline{CAS} is at V_{IH} , the input is in a High Impedance (Hi-Z) state. The D_{OUT} states for various operating modes are shown in Table 3. For a Read or Read-Modify-Write cycle, D_{OUT} will maintain in the Hi-Z state until the data is valid whereupon it will go to V_{CC} or V_{SS} , depending upon the data.

For an "Early" Write cycle, D_{OUT} remains in the Hi-Z state which allows "wire-OR" of D_{IN} and D_{OUT} . D_{OUT} is indeterminate for the period between an "Early" Write ($t_{WCS} = 0$) and a Read-Modify-Write cycle ($t_{RWD} > t_{RWD \min}$ and $t_{CWD} > t_{CWD \min}$). A \overline{RAS} -only refresh cycle or a \overline{CAS} -only cycle will have no effect on

the D_{OUT} state which will remain in the Hi-Z state. The D_{OUT} remains valid from access time until \overline{CAS} goes high. Holding \overline{CAS} low and taking \overline{RAS} high will not affect the state of the D_{OUT} . The D_{OUT} remains valid following a valid Read cycle regardless of the number of subsequent \overline{RAS} -only cycles performed on the device up to the $t_{CAS \max}$ limit. These secondary \overline{RAS} cycles are \overline{RAS} -only refresh cycles to the 2118.

Table 3. Intel® 2118 Data Output Operation for Various Types of Cycles

Type of Cycle	Data Output State
Read Cycle	Data from Addressed Memory Cell
Early Write Cycle	Hi-Z
\overline{RAS} -Only Refresh Cycle	Hi-Z
\overline{CAS} -Only Cycle	Hi-Z
Read/Modify/Write Cycle	Data from Addressed Memory Cell
Delayed Write Cycle	Indeterminate
Hidden Refresh Cycle	Data from Addressed Memory Cell
Page Mode Read Cycle (Entry or Internal Cycle)*	Data from Addressed Memory Cell
Page Mode Write Cycle (Entry or Internal Cycle)*	Hi-Z
Page Mode R/M/W Cycle (Entry or Internal Cycle)*	Data from Addressed Memory Cell

*The entry cycle is the first cycle of the page and the internal cycles are the subsequent cycles of the page operation.

3.13 Power-on Current

Power-on current is characteristic of MOS devices which contain on-chip substrate bias generators. All MOS transistor thresholds are a function of the substrate bias. The more negative the substrate voltage, the higher the device threshold. Without substrate bias, all transistors in the 2118 would be on, resulting in relatively large I_{DD} requirements for the power supply. A sufficient power supply current margin must exist under power-on conditions. A marginal power supply might not be capable of supplying enough current to achieve full voltage of V_{DD} .

A system power supply, designed such that a majority of devices are assumed to be in the refresh/standby mode (about 2.5 mA per device), may not be capable of providing sufficient current during power-on. The limiting criterion for the power-on current is the state of \overline{RAS} and \overline{CAS} . As shown in Figure 18, the current required for power-on is significant. If the device is powered up with \overline{RAS} and \overline{CAS} high or tracking V_{DD} , the devices are deselected, and the internal circuitry is disabled. Figure 18a shows that the current will be less than the specified standby I_{DD} current maximum.

However, if the device is powered up with $\overline{\text{RAS}}$ and $\overline{\text{CAS}}$ held at V_{SS} (Figure 18b), the power-on current is several times that of the specified I_{DD} standby (about 5 mA typically).

Figure 18b also demonstrates the activation point of the substrate bias generator. At about 2V, the generator begins to operate, causing the V_{DD} current to decrease. There are at least two ways of designing the memory system array to eliminate this sensitivity. One way is to provide V_{DD} pull-up resistors on the $\overline{\text{RAS}}$ and $\overline{\text{CAS}}$ lines. The other method would be to power-on the TTL logic prior to power-on of the memory array. The first method is simple, easy to implement, and relatively low in cost.

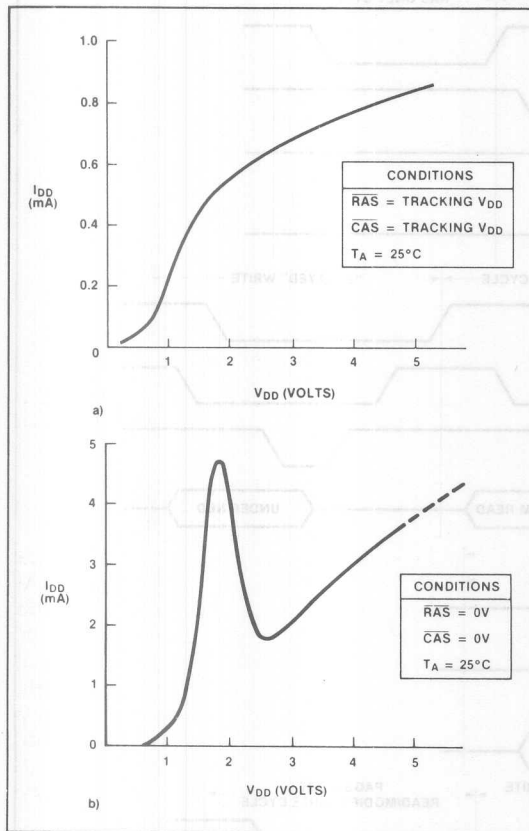


Figure 18. Intel® 2118 Typical Power-On Current

4. DATA CYCLES/TIMING

A memory cycle begins with a negative transition of $\overline{\text{RAS}}$. Both the $\overline{\text{RAS}}$ and $\overline{\text{CAS}}$ clocks are TTL compatible. The 2118 input buffers convert the TTL level signals to MOS levels inside the device. Therefore, the delay associated with external TTL-MOS level converters is not added to the 2118 system access time.

$\overline{\text{RAS}}$ and $\overline{\text{CAS}}$ have minimum pulse widths as specified in the 2118 Data Sheet. These minimum pulse widths

must be maintained for proper device operation and data integrity. A cycle, once begun, must be within specification.

Figure 19 briefly summarizes the various active cycles which follow.

4.1 Read Cycle

A Read cycle is performed by maintaining $\overline{\text{WE}}$ high during a $\overline{\text{RAS}}/\overline{\text{CAS}}$ operation. The output pin of a selected device remains in a high impedance state until valid data appears at the output at access time.

Device access time, t_{ACC} , is the longer of two calculated intervals:

$$\text{Eq. (1)} \quad t_{\text{ACC}} = t_{\text{RAC}} \text{ or}$$

$$\text{Eq. (2)} \quad t_{\text{ACC}} = t_{\text{RCD}} + t_{\text{CAC}}$$

Access time from $\overline{\text{RAS}}$ and t_{RAC} , and access time from $\overline{\text{CAS}}$ and t_{CAC} , are device parameters. Row to column address strobe delay time, t_{RCD} , is a system-dependent timing parameter. For example, substituting the device parameters of the 2118-4 yields:

$$\text{Eq. (3)} \quad t_{\text{ACC}} = t_{\text{RAC}} = 120 \text{ ns for } 25 \text{ ns} \leq t_{\text{RCD}} \leq 55 \text{ ns}$$

or

$$\text{Eq. (4)} \quad t_{\text{ACC}} = t_{\text{RCD}} + t_{\text{CAC}} = t_{\text{RCD}} + 65 \text{ ns for } t_{\text{RCD}} > 55 \text{ ns}$$

Note that if $25 \text{ ns} \leq t_{\text{RCD}} \leq 55 \text{ ns}$, device access time is determined by equation 3 and is equal to t_{RAC} . If $t_{\text{RCD}} > 55 \text{ ns}$, access time is determined by equation 4. This 30 ns interval (shown in the t_{RCD} inequality in equation 3), in which the falling edge of $\overline{\text{CAS}}$ can occur without affecting access time, allows for system timing skew in the generation of $\overline{\text{CAS}}$. This allowance for t_{RCD} skew is designed in at the device level to provide for the fastest access times to be utilized in practical system designs.

4.2 Write Cycle (Early Write)

A Write cycle is performed by bringing $\overline{\text{WE}}$ low before $\overline{\text{CAS}}$. D_{IN} is written into the selected bit. D_{OUT} remains in the Hi-Z state.

4.3 Read-Modify-Write Cycle (Delayed Write)

A Read-Modify-Write cycle (R-M-W cycle) is performed by bringing $\overline{\text{WE}}$ low with $\overline{\text{RAS}}$ and $\overline{\text{CAS}}$ low. In a R-M-W cycle, D_{OUT} is data read and does not change during the Modify-Write portion of the cycle. In a Delayed Write cycle, where timing considerations are not met for R-M-W cycles, D_{OUT} is indeterminate.

In any type of Write cycle D_{IN} must be valid at or before the falling edge of $\overline{\text{WE}}$ or $\overline{\text{CAS}}$ whichever is latest.

4.4 $\overline{\text{CAS}}$ -Only Cycle

A CAS-only cycle has no effect on the 2118. The 2118 remains in the lowest power, standby condition.

4.5 Refresh Cycle

A cycle at each of the 128 row addresses (A_0 through A_6) will refresh all storage cells. Any memory cycle—Read, Write (Early Write, Delayed Write, R-M-W) or $\overline{\text{RAS}}$ -only—refreshes the bits selected by the $\overline{\text{RAS}}$ addresses.

4.5.1 READ CYCLE REFRESH

This refresh mode is useful only when the memory system consists of single row devices. When used with more than one row of devices, output bus contention will result.

4.5.2 WRITE CYCLE REFRESH

A Write cycle will perform a refresh. However, the selected cell will undergo a Write. This will cause a change of state of selected cell, while the other 127 cells are refreshed.

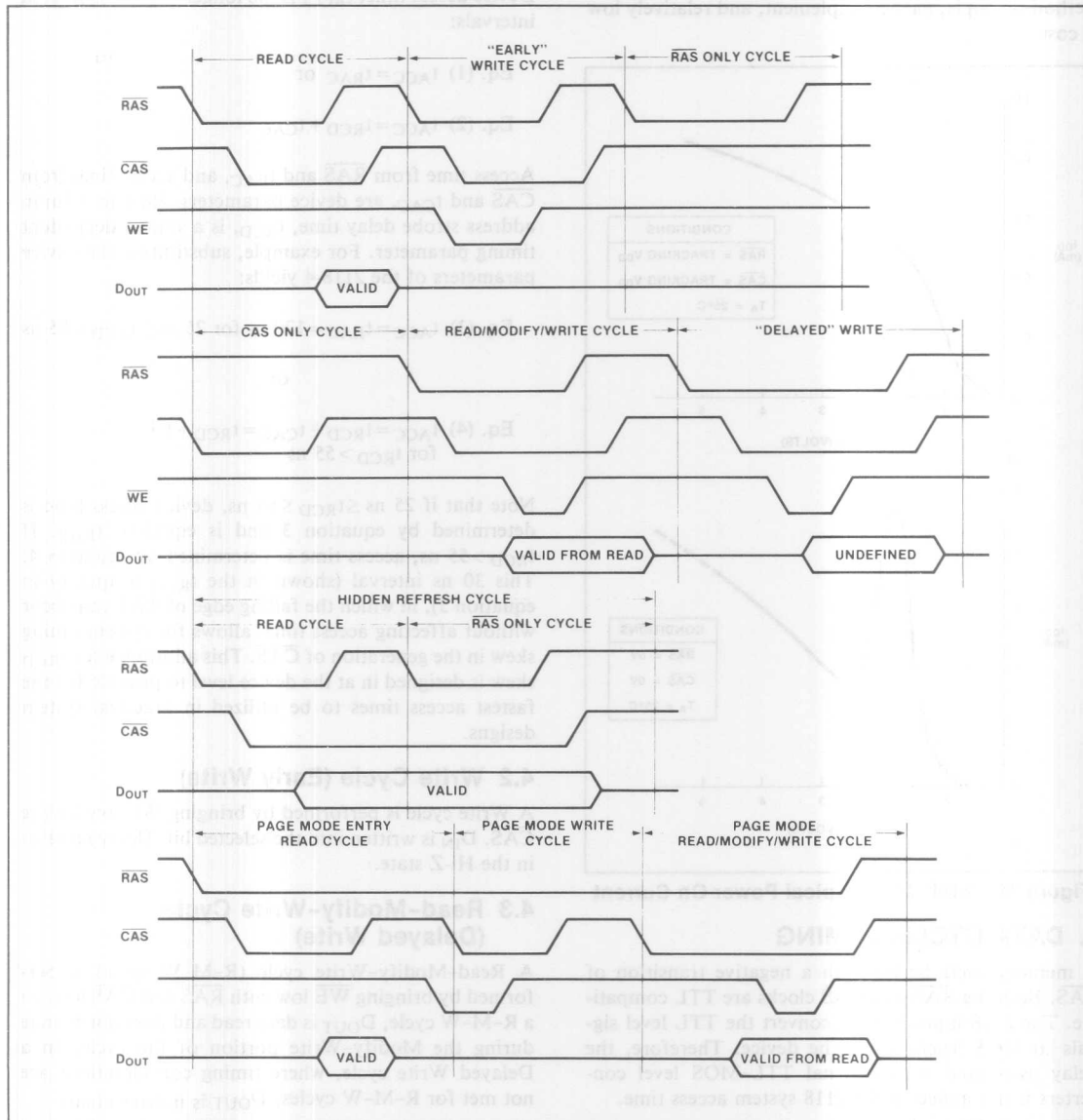


Figure 19. Intel® 2118 Operation of Data Output for Various Active Cycles

For an Early Write refresh cycle, there will be no output bus contention since the output remains in the Hi-Z state. For Delayed Write or R-M-W refresh cycles involving more than one row of devices, bus contention will result.

4.5.3 $\overline{\text{RAS}}$ -ONLY REFRESH

A cycle with $\overline{\text{RAS}}$ active refreshes the 2118. This is the recommended refresh mode, especially when the memory system consists of multiple rows of memory devices. The D_{OUT} s may be OR-tied with no bus contention when $\overline{\text{RAS}}$ -only refresh cycles are performed.

4.5.4 HIDDEN $\overline{\text{RAS}}$ -ONLY REFRESH

The 2118 is designed for "hidden" refresh operation. Hidden refresh accomplishes a refresh cycle following a Read cycle without disturbing the D_{OUT} . Once valid, D_{OUT} is controlled solely by $\overline{\text{CAS}}$. After a Read cycle, $\overline{\text{CAS}}$ is held low while $\overline{\text{RAS}}$ goes high for precharge. A $\overline{\text{RAS}}$ -only cycle is then performed and D_{OUT} remains valid. However, for operation in this mode $\overline{\text{CAS}}$ must be decoded along with $\overline{\text{RAS}}$ for the Read and Write cycles. $\overline{\text{CAS}}$ cannot be driven as a common clock to the entire array since it would cause devices being refreshed only to interpret this operation as a $\overline{\text{RAS}}/\overline{\text{CAS}}$ cycle.

4.6 Page Mode Operation

Page Mode operation allows additional columns of the selected device to be accessed for a given row address set. This is done by maintaining $\overline{\text{RAS}}$ low while successive $\overline{\text{CAS}}$ cycles are performed.

Page Mode operation allows a maximum data transfer rate as $\overline{\text{RAS}}$ addresses are maintained internally and do not have to be re-applied. During this operation, Read, Write and R-M-W cycles are possible. Following the entry cycle into Page Mode operation, access is t_{CAC} dependent. The Page Mode cycle is dependent upon $\overline{\text{CAS}}$ pulse width (t_{CAS}) and the $\overline{\text{CAS}}$ precharge period (t_{CP}).

5. SYSTEM DESIGN CONSIDERATIONS

5.1 Power Calculations

Because of 5V operation and low current requirements, the 2118 consumes very little power—less than a third that required by a comparable 12V device.

Calculating total 2118 power consumption is a simple matter. For the purpose of performing this power calculation, an example system is assumed (organized as 64K words \times 16 bits). The first step is to find total 2118 current by summing the three individual V_{DD} supply currents: operating current (I_{DDO}), standby current (I_{DDS}), and refresh current (I_{DDR}). Total 2118 power consumption equals the total 2118 current multiplied by the maximum supply voltage (V_{DD}). Total system power consumption is determined by adding the support circuitry power requirements to the total 2118 power.

Examples of these calculations, along with a power/bit determination, are presented in following sections.

5.1.1 OPERATING CURRENT (I_{DDO})

Active operating current is determined by the following equation:

$$\text{Eq. (1)} \quad I_{\text{DDO}} = (I_{\text{DD2}} + I_{\text{DDLO}})K$$

Where: I_{DDO} = the operating V_{DD} supply current.

K = the number of active devices (selected at one time by both $\overline{\text{RAS}}$ and $\overline{\text{CAS}}$).

I_{DDLO} = the 2118 output load current (output leakage current plus the load devices input current). For example, if four devices are on the output line the output leakage current is the sum of the input current (I_{IN}) for the load plus the three leakage currents (I_{LO}) for the three devices in standby.

5.1.2 STANDBY CURRENT (I_{DDS})

Standby current is determined by the following equation:

$$\text{Eq. (2)} \quad I_{\text{DDS}} = I_{\text{DD1}} \times M$$

Where: I_{DD1} = the V_{DD} supply current.

M = the number of inactive devices (not selected by $\overline{\text{RAS}}$; receiving $\overline{\text{CAS}}$ -only cycles).

5.1.3 REFRESH CURRENT (I_{DDR})

Refresh current is determined by the following equation:

$$\text{Eq. (3)} \quad I_{\text{DDR}} = (I_{\text{DD3}} \times N) (t_{\text{RC}}/t_{\text{REF}}) \quad (128)$$

Where: I_{DD3} = the V_{DD} supply current, $\overline{\text{RAS}}$ -only cycle.

N = the total number of devices in the system.

t_{RC} = the refresh cycle time.

t_{REF} = the time between refresh cycles.

Since I_{DD3} is not a full-time current, the fraction t_{RC} over t_{REF} represents the duty cycle for one address. There are 128 row addresses receiving refresh, so the duty cycle is multiplied by 128.

5.1.4 TOTAL 2118 POWER

Total 2118 power equals the sum of the three currents multiplied by the worst case supply voltage. This is expressed by the following equation:

$$\text{Eq. (4)} \quad \text{Power} = (I_{\text{DDO}} + I_{\text{DDS}} + I_{\text{DDR}}) V_{\text{DD}} (\text{max})$$

apply:

$N = 64$ device in system

$K = 16$ device active at one time

$M = N - K$ device in standby

$= 64 - 16$

$= 48$

Referring to the Intel 2118 Data Sheet¹ and the Intel 8282 Data Sheet² we obtain the following values:

$I_{DD1} = 2 \text{ mA}$ 2118-7, $t_{REF} = 2 \text{ ms}$

$I_{DD2} = 23 \text{ mA}$ 2118-7, $t_{RC} = 320 \text{ ns}$

$I_{DD3} = 14 \text{ mA}$ 2118-7

$I_{LO} = 10 \text{ } \mu\text{A}$ 2118-7

$I_{IN} = 200 \text{ } \mu\text{A}$ 8282

To calculate I_{DDO} :

$$\begin{aligned} \text{Eq. (1) } I_{DDO} &= (I_{DD2} + I_{DDLO})K \\ &= \{23 \text{ mA} + [3(10 \text{ } \mu\text{A}) + 200 \text{ } \mu\text{A}]\}16 \\ &= 371.68 \text{ mA} \end{aligned}$$

To calculate I_{DDS} :

$$\begin{aligned} \text{Eq. (2) } I_{DDS} &= (I_{DD1})M \\ &= (2 \text{ mA})48 \\ &= 96 \text{ mA} \end{aligned}$$

To calculate I_{DDR} :

$$\begin{aligned} \text{Eq. (3) } I_{DDR} &= (I_{DD3} \times N) (t_{RC}/t_{REF}) (128) \\ &= (14 \text{ mA} \times 64) \frac{320 \text{ ns}}{2 \text{ ms}} (128) \\ &= (896 \text{ mA}) (0.02) \\ &= 18.35 \text{ mA} \end{aligned}$$

To calculate total power:

$$\begin{aligned} \text{Eq. (4) Power} &= (I_{DDO} + I_{DDS} + I_{DDR}) V_{DD} (\text{max}) \\ &= (371.7 \text{ mA} + 96 \text{ mA} + 18.4 \text{ mA}) \\ &\quad 5.5\text{V} \\ &= 2.7 \text{ watts} \end{aligned}$$

The power/bit is equal to:

$$\begin{aligned} \text{Power/Bit} &= \text{Total 2118 Power/Number of Devices} \\ &\quad \times \text{Bits per Device} \\ &= 2.7 / (64 \times 16,384) \\ &= 2.6 \text{ } \mu\text{watts/bit} \end{aligned}$$

¹Intel® 2118 Family 16,384 × 1-Bit Dynamic RAM, July 1979

²Intel® Component Data Catalog

AP-75

board area while yielding wider power supply and timing operating margins for increased reliability and easier manufacture. The key areas of consideration are:

- 1) Ground (V_{SS}) and power (V_{DD}) gridding
- 2) Memory array/control line routine
- 3) Control logic centralization
- 4) Power supply decoupling

5.2.1 GROUND AND POWER GRIDGING

Ground and power gridding can contribute to excess noise and voltage drops if not properly structured. An example of an unacceptable method is presented in Figure 20. This type of layout promotes accumulated transient noise and voltage drops for the device located at the end of each trace (path).

Transient effects can be minimized by adding extra circuit board traces in parallel to reduce interconnection inductance (Figure 21).

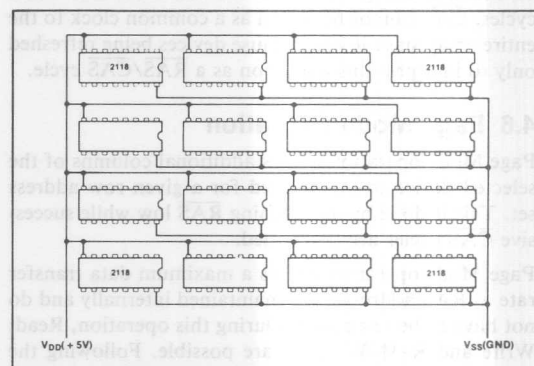


Figure 20. Unacceptable Power Distribution

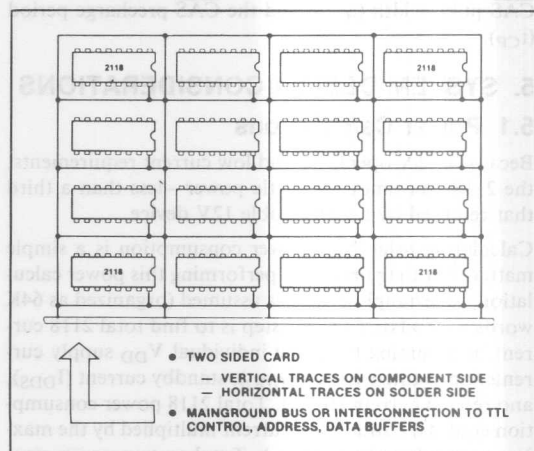


Figure 21. Recommended Power Distribution—Gridding

Address lines need to be kept as short and direct as possible. The lone serpentine line depicted in Figure 22 should be avoided, since the devices furthest away from the driver will receive a valid address at a later time than the closer ones. A better way to route address lines is in

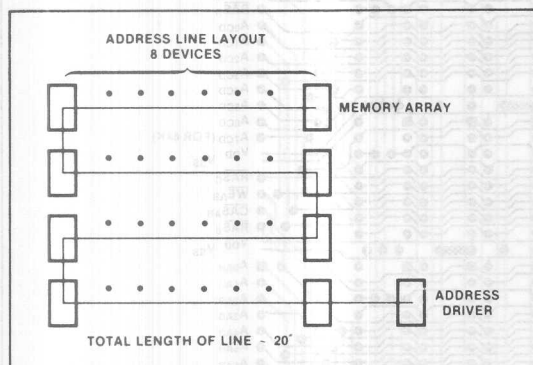


Figure 22. Unacceptable Address Line Routing (Serpentine)

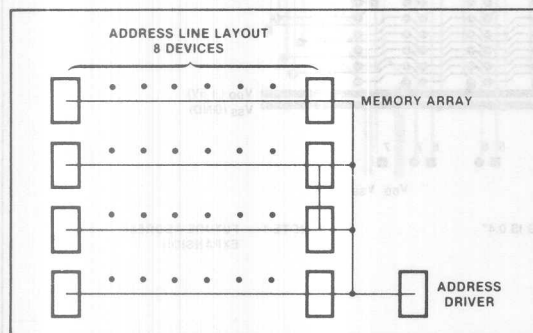


Figure 23. Recommended Address Line Routing

gether from a centralized board area will also minimize skew.

5.2.3 CONTROL LOGIC CENTRALIZATION

Memory control logic should be strategically located in a centralized board position to reduce trace lengths to the memory array. Long trace lines are prone to ringing and capacitive coupling, which can cause false triggering of timing circuits. Short lines minimize this condition and also result in less system skew.

A practical memory array layout is presented in Figure 24. Typically, this pattern and its "mirror image" are placed on each side of the memory control logic for a practical memory board design.

5.2.4 POWER SUPPLY DECOUPLING

For best results, decoupling capacitors are placed on the memory array board in a checkerboard arrangement (Figure 24). High frequency 0.1 μ F ceramic capacitors are the recommended type. In this arrangement each memory is effectively decoupled by a "half capacitor" and noise is minimized because of the low impedance across the circuit board traces. Typical V_{DD} noise levels for this array are less than 300 mV.

A large tantalum capacitor (typically one 47 μ F per 64 devices) is required at the circuit board edge connector power input pins to recharge the 0.1 μ F capacitors between memory cycles.

6. SUMMARY

The Intel® 2118, made possible by exclusive Intel HMOS technology, introduces a new generation of dynamic RAM devices, featuring +5V only, TTL compatible operation, high performance, low power, and ease of use.

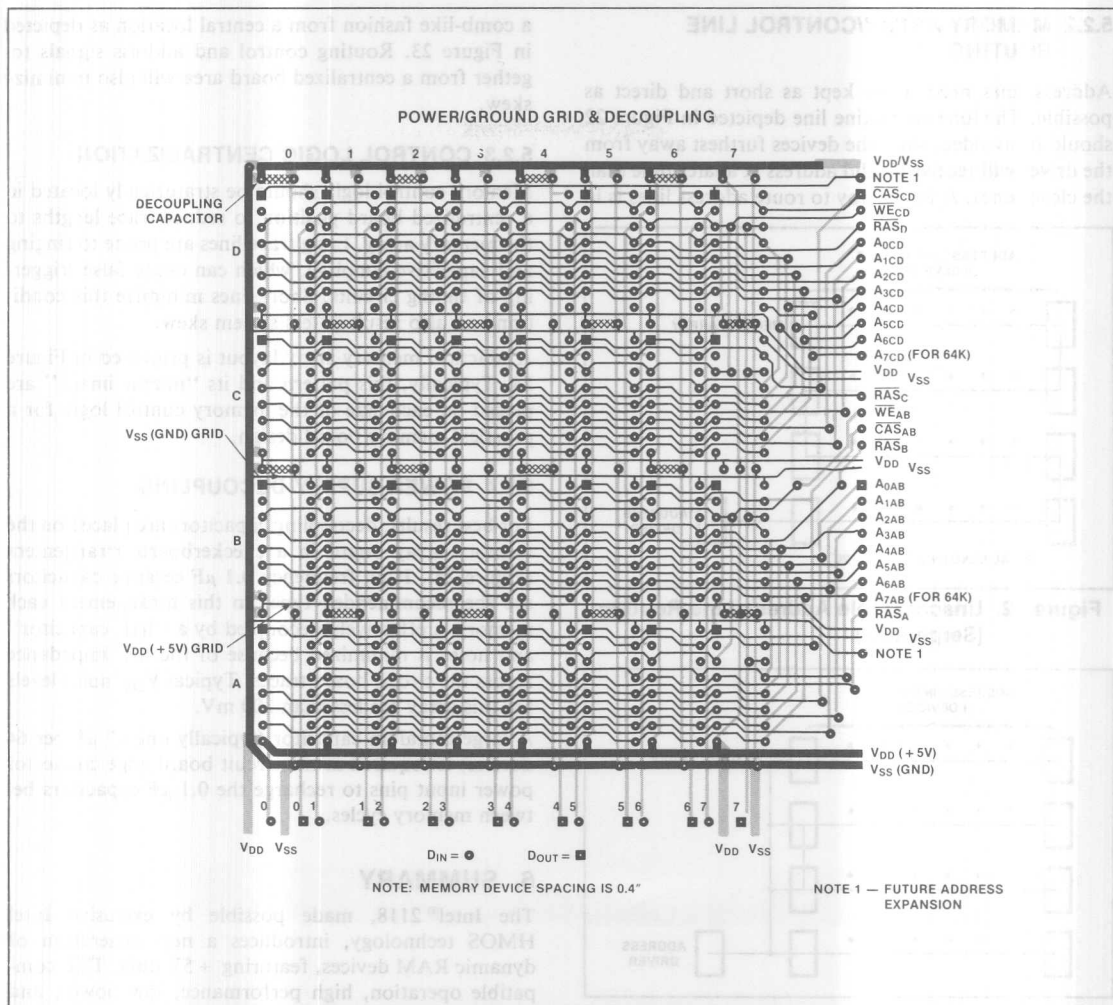


Figure 24. 2118/2164 Memory Array P.C.B. Layout

Programmable Read Only Memories (PROMs)

3

CHAPTER 3

and 16K Bipolar PROM has established new levels of reliability for the PROM family. High temperature life-testing was used to evaluate the long-term failure rate. At 55°C and 60% confidence level, a failure rate of 0.005%/1000 hours was calculated. Detailed test results are included in this report.

RELIABILITY TESTING AND RESULTS

Four categories of testing were used to assure the electrical reliability of the Advanced PROM family:

1. High Temperature Dynamic Lifetest
2. High Temperature Reverse Bias
3. High Temperature Storage
4. Temperature Cycling

High Temperature Dynamic Lifetest

This test is used to accelerate failure mechanisms by operating the devices at an elevated temperature of 125°C. The data obtained are translated to a lower temperature using the Arrhenius Plot in Figure 1 giving a larger number of equivalent hours of test. During the test the memory is sequentially addressed and the outputs are exercised, but not monitored or loaded. Results of lifetesting on 3625A and 3636 are shown in Table I along with failure analysis. In order to best determine long-term failure rates all devices used for lifetesting and High Temperature Reverse Bias (HTRB) are subjected to standard Intel screening plus a 48 hr. burn-in to eliminate infant mortality. Results from the burn-in are also shown in Table I.

Failure rate calculations are shown in Table II for each device type. Failure rate calculations are made using the appropriate activation energy¹ and the Arrhenius Plot in Figure 1. The total equivalent device hours at a given temperature can thus be determined. The failure rate is then calculated by dividing the number of failures by the equivalent device hours and is expressed as a %/1000 hours. The failure rate is adjusted by a negative factor related to the number of device hours using a chi-square distribution to arrive at a confidence-level-associated failure rate.

Combining the device hours for all PROMs on this new process gives a reliability indicator of the technology. This calculation is shown in Table II. A failure rate of 0.009%/1000 hours at 70°C and 0.005%/1000 hours at 55°C using 60% confidence level are determined for the 3625A and 3636.

¹Since no lifetest failures were encountered, the standard bipolar activation energy of 0.4 eV was used.

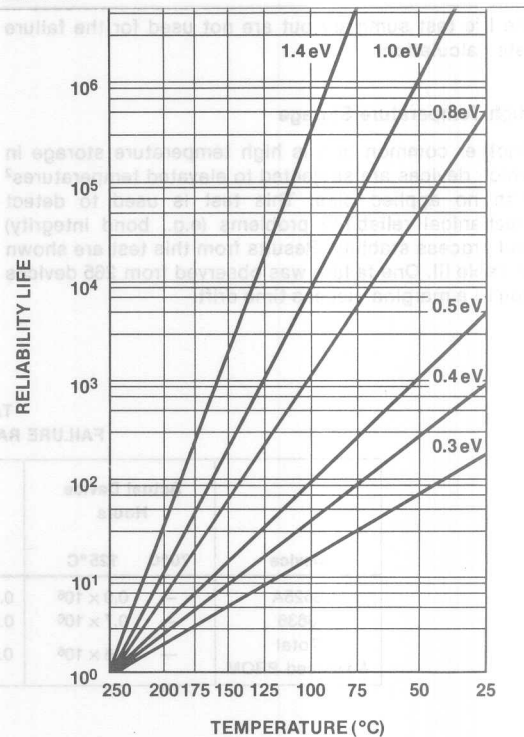


Figure 1. Arrhenius Plot

PRELIMINARY LIFE TIME TESTING CONTINUING

TABLE I

Device	125°C Dynamic Lifetest					150°C HTRB	
	48	168	500	1K	2K	500	1K
3625A	0/451	0/285	0/285	0/285	—	0/91	0/91
	0/156	0/84	0/84	0/84	—	0/25	0/25
	0/118	0/70	0/45	0/45	—	0/25	0/25
	0/101	0/101	0/101	0/101	—	—	—
	0/100	0/100	—	—	—	—	—
	0/202	0/202	0/141	0/141	—	0/20	0/20
	0/167	0/107	0/107	0/107	—	0/20	0/20
	0/101	0/101	0/100	0/100	—	0/20	0/20
	0/96	0/66	0/66	0/66	—	0/10	0/10
	0/127	0/87	0/87	0/87	—	0/20	0/20
3636	1/127 ^A	0/84	0/77	0/77	—	0/20	0/20
	1/164 ^B	0/103	0/103	0/103	—	0/20	0/20
	1/449 ^C	0/200	0/200	0/200	0/200	—	—
	0/80	0/80	—	—	—	—	—
	0/85	0/85	—	—	—	—	—
	0/554	0/352	0/192	—	—	—	—
	1/314 ^D	0/160	—	—	—	—	—
	—	—	—	—	—	—	—

A = A.C. Degradation
B = Multi-row leakage

C = Single bit
D = Single column failure

High Temperature Reverse Bias (HTRB)

This test is performed at 150°C and is effective in testing for leakage failures and device parameter drift. High Temperature Reverse Bias results are included in the life test summary but are not used for the failure rate calculation.

High Temperature Storage

Another common test is high temperature storage in which devices are subjected to elevated temperatures² with no applied bias. This test is used to detect mechanical reliability problems (e.g., bond integrity) and process stability. Results from this test are shown in Table III. One failure was observed from 265 devices due to a marginal access time drift.

Temperature Cycling

This test consists of cycling the temperature of the chamber housing the devices from -65°C to 150°C. This test is used to detect mechanical reliability problems and microcracks. Results are also shown in Table III. No rejects were found on 125 devices.

2250°C for hermetic packages.

TABLE II
FAILURE RATE PREDICTIONS

Device	Actual Device Hours		Equivalent Device Hours			Fail Rate/1000 Hr. (60% Confidence Level)		
	70°C	125°C	Ea	55°C	70°C	#Fail	55°C	70°C
3625A	—	0.9×10^6	0.4 eV	10.8×10^6	5.7×10^6	0	0.009%	0.017%
3636	—	0.7×10^6	0.4 eV	8.4×10^6	4.5×10^6	0	0.01%	0.02%
Total Advanced PROM	—	1.6×10^6	0.4 eV	19.2×10^6	10.1×10^6	0	0.005%	0.009%

TABLE III

Device	250°C Bake			Temperature Cycling (200 Cycles)	
	168	500	1K	200 Cycles	
3625A	0/50	0/50	0/50	0/50	
	0/20	0/20	0/20	0/20	
	0/20	0/20	0/20	0/20	
	0/20	0/20	0/20	0/20	
	0/10	0/10	0/10	0/10	
3636	1/20 ^A	0/19	0/19	—	
	0/20	0/20	0/20	—	
	0/20	0/20	0/20	0/20	

A = Access time degradation

APPLICATION OF THE 2716 16K EPROM

INTRODUCTION

The INTEL® 2716 is a fully static 16,384-bit (2048 x 8) Erasable Programmable Read Only Memory, or EPROM. The device is packaged in a standard 24-pin DIP, which has a transparent lid to allow erasure in a manner similar to that of the INTEL® 1702A and 2708. Maximum access time is 450ns. The device requires a single power supply ($V_{CC} = 5V \pm 5\%$) for normal read cycles; during programming the program power supply (V_{pp}) must be raised to +25V to program each location, a single TTL level pulse is required; one 50ms pulse per address programs 8 bits in parallel. The addresses can be randomly programmed.

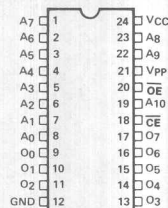
All input signals are fully TTL compatible during both the read and program modes. The data outputs are three state to facilitate memory expansion by OR tying. Initially and after each erasure the 2716 contains all TTL highs ("1"s); programming or introducing TTL lows ("0"s) is accomplished by: 1) raising the V_{pp} pin from +5V to +25V, 2) applying TTL level addresses and TTL level data, 3) raising the \overline{CS} pin to a TTL high, and 4) applying a single 50ms TTL level pulse to the PD/PGM input.

The V_{pp} supply may be left at the +25V level for program verification, but should be returned to +5V level during normal read cycles to reduce power dissipation.

DEVICE DESCRIPTION

The 2716 is packaged in an industry standard 24 pin DIP as shown in Figure 1. The functions of the various control pins are shown in Table I.

During read operation \overline{CS} is used to select and deselect the 2716. The PD/PGM pin is maintained at



PIN NAMES	
A0-A10	ADDRESSES
\overline{OE}	OUTPUT ENABLE
\overline{CE}	CHIP ENABLE
O0-O7	OUTPUTS

Figure 1. 2716 Pin Configuration.

Table I. 2716 Pin Connections and Functions.

MODE	\overline{CE} (18)	\overline{OE} (20)	V_{pp} (21)	V_{CC} (24)	OUTPUTS (9-11, 13-17)
Read	V_{IL}	V_{IL}	+5	+5	D_{OUT}
Deselect	Don't Care	V_{IH}	+5	+5	High Z
Power Down	V_{IH}	Don't Care	+5	+5	High Z
Program	Pulsed V_{IL} to V_{IH}	V_{IH}	+25	+5	D_{IN}
Program Verify	V_{IL}	+25	+5	+5	D_{OUT}
Program Inhibit	V_{IL}	V_{IH}	+25	+5	High Z

V_{IL} , while V_{pp} , the program power supply, is maintained at +5V. As shown in the D.C. Device Characteristics Section, I_{pp1} (the current required by pin 21) is 5mA maximum during read mode, so pin 20 should be kept at V_{CC} except when programming. As a convenience to users, it is allowable to keep the V_{pp} pin at +25 volts for program verification, but it must be returned to +5V upon completing program verification. This is easily accomplished by connecting a diode from pin 24 to pin 21 as shown in Figure 2. The tolerance on V_{pp} allows for a diode drop as discussed in the D.C. Operating Characteristics section. For read only applications, the V_{pp} pin may be tied directly to the V_{CC} pin.

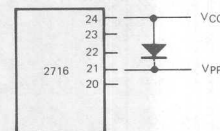
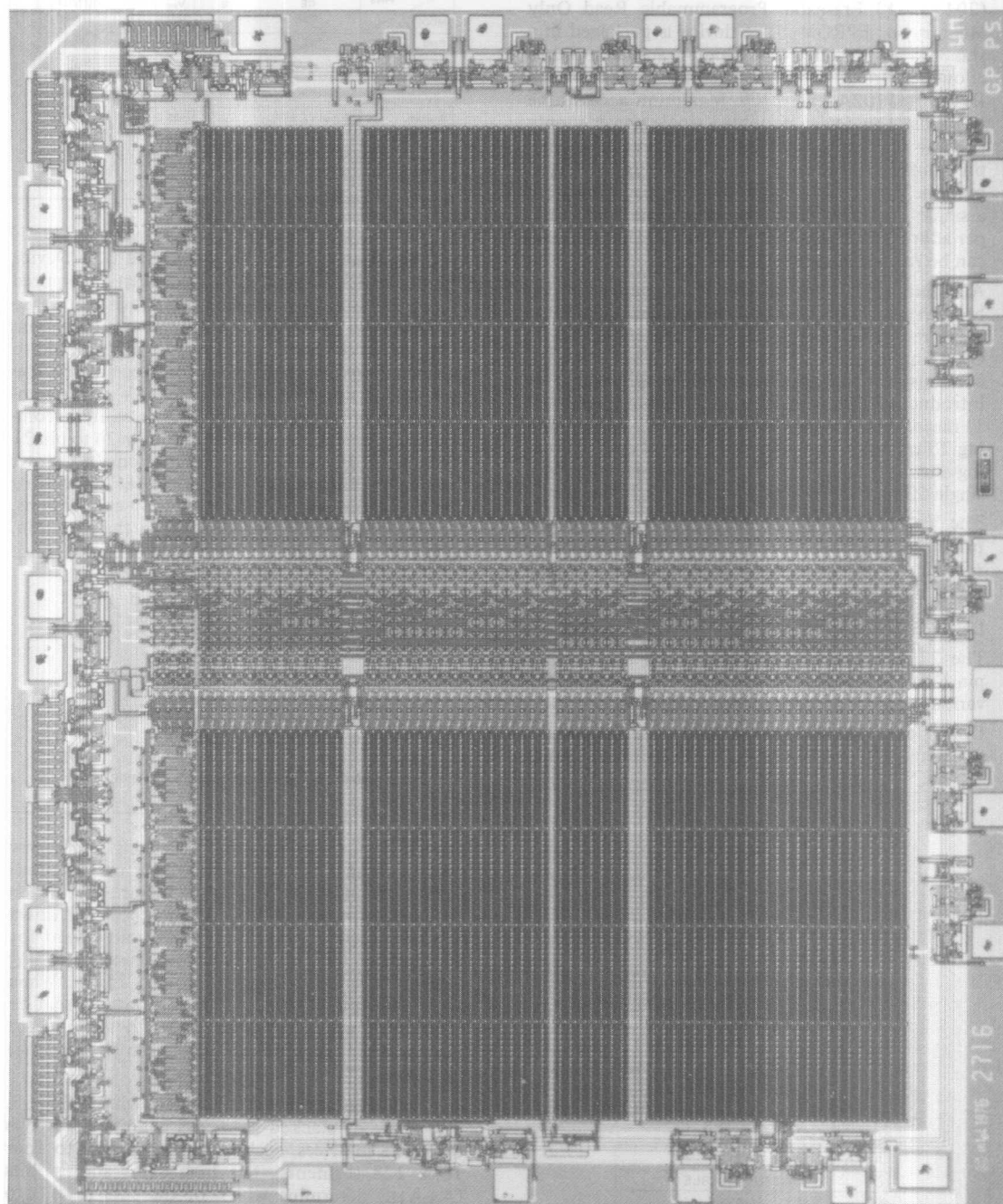


Figure 2. 2716 Power Supply Connections.

The \overline{CE} input serves several functions. When low this signal enables the address, data and \overline{CS} input buffers, whether V_{pp} is at +25V or +5V. When high with V_{pp} at +5V, the 2716 is powered down and the outputs are deselected without regard for the state of \overline{OE} . In this mode the maximum I_{CC} current is reduced from 100mA to 25mA. When \overline{OE} is high and V_{pp} is at 25V, the data present on the output will be programmed into the selected address when \overline{CE} is pulsed high (from V_{IL} to V_{IH}) for 50ms.

A block diagram for the 2716 is shown in Figure 3. The array of stacked gate cells is arranged as two 64 x 128 matrices, each of which is split into four 16 x 128 segments. The high order address bits (A4-A10) determine which of the 128 rows is to be accessed by way of the top select gate, while the low order address bits (A0-A3) perform the column decode function by activating the 1 of 16 decoders which are associated with each output bit.



Photomicrograph of the Intel 2716 16K (2048 X8) MOS EPROM

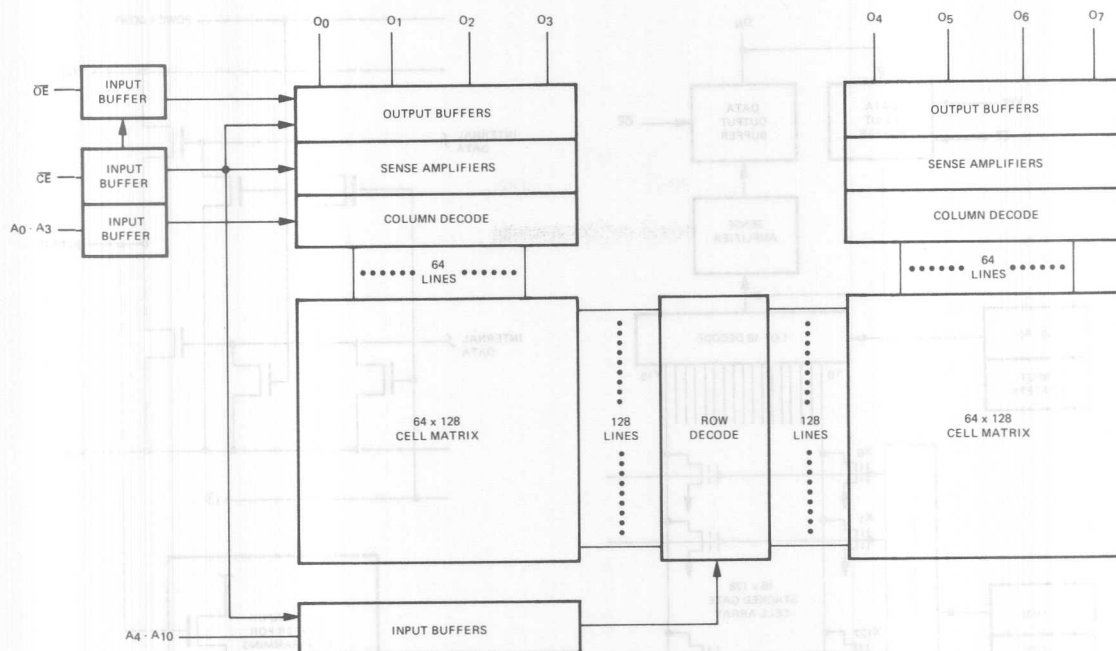


Figure 3. Detailed Block Diagram.

Cell Description

The heart of the 2716 is the single transistor stacked gate cell, which is similar to the cell used in the INTEL® 2708. The cell consists of a floating gate, used to store charge, and a top select gate which is connected to the output of the row decoder. The cell is programmed by injection of high energy electrons through the isolating oxide and onto the floating gate. Once there, the charge is trapped, as there are no electrical connections to the floating gate. The presence of electrons on the floating gate causes a shift in cell threshold, as shown in Figure 4. In the initial or erased state the threshold of the cell is low, selection via the top gate will cause the column line to discharge, which is sensed as a "HIGH" by the sense amplifier. Programming shifts the threshold to a higher level, and selection of the cell will not turn it on, the column line will not discharge, and a low will be sensed by the sense amplifier. The status of the cell is determined by examining its state at the sense threshold; if the cell is erased (HIGH data) selection will cause a higher current to flow between the source and drain than if the cell is programmed (LOW data).

Memory Array Operation

The cells described in the previous paragraph are interconnected to form a split 128 x 128 cell ma-

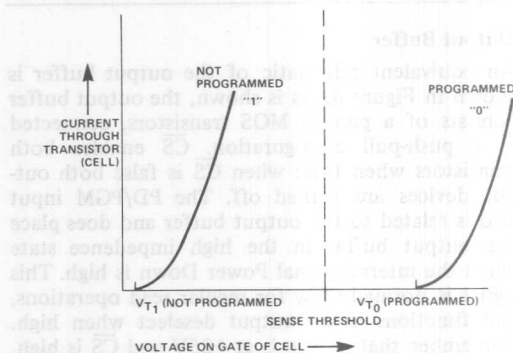


Figure 4. Storage Cell Threshold Shift

trix, as shown in Figure 3. This array is divided into 8 sections organized as 16 x 128 cells. Each of these sections is connected to a column decoder, which selects one of 16 columns, connecting it to the sense amplifier which is associated with the particular bit. The sense amplifier is directly connected to the output buffer associated with the same bit. This data flow is illustrated graphically in Figure 5.

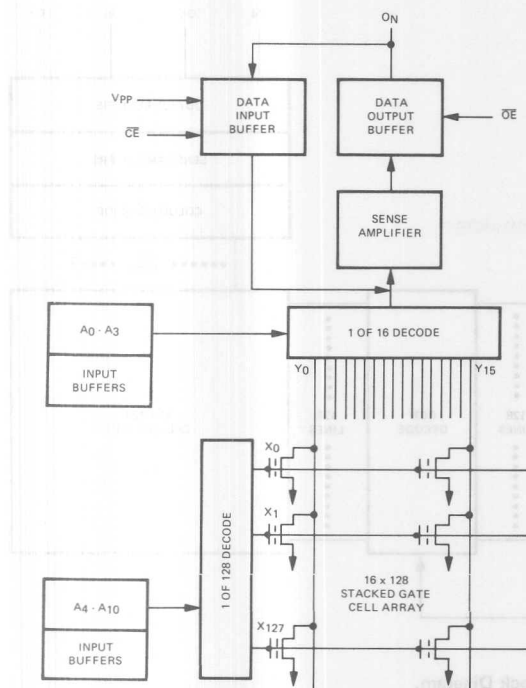


Figure 5. 2716 Single Bit Data Flow.

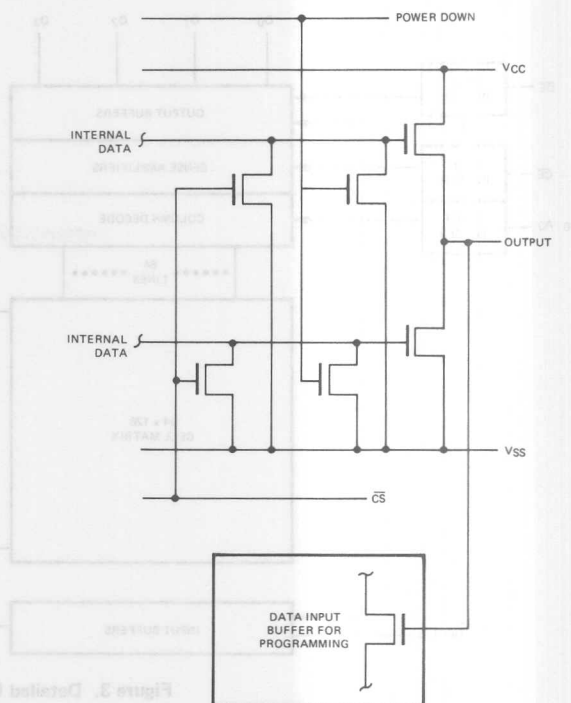


Figure 6. 2716 Output Buffer.

Output Buffer

An equivalent schematic of the output buffer is shown in Figure 6. As is shown, the output buffer consists of a pair of MOS transistors, connected in a push-pull configuration. \overline{CS} enables both transistors when true; when \overline{CS} is false both output devices are turned off. The PD/PGM input also is related to the output buffer and does place the output buffer in the high impedance state when the internal signal Power Down is high. This signal is normally low for regular read operations, and functions as an output deselect when high. Remember that if V_{pp} is at +25V and \overline{CS} is high, raising PD/PGM high will cause a program cycle on the selected address.

READ MODE

The 2716 requires only one power supply, +5V. The device is rated to meet all applicable specifications with this supply held within $\pm 5\%$ of its nominal value. The Absolute Maximum Ratings in the data sheet are the maximum that the various device parameters can withstand and should not be exceeded during any phase of device operation, including programming.

D.C. Characteristics

Only those D.C. Characteristics that require special attention by the user are presented in this section.

The reader is referred to the 2716 device data sheet for further details. The pertinent D.C. device specifications are tabulated in Table II.

The range of the leakage currents shown in Table II apply for all inputs and outputs, including the outputs (00-07) when they are serving as data inputs for programming.

I_{pp1} is the current required by the V_{pp} pin (pin 21) when the V_{pp} supply is set to 5V, as it would be for normal read operations. The device specification requires a $\pm 5\%$ tolerance on the V_{CC} supply. In anticipation that users will couple pin 21 to pin 24 by way of a diode, the tolerance on V_{pp} has been relaxed to $\pm 0.6V$ to allow for the forward drop of the diode.

I_{pp} is only applicable to the current drawn by pin 21 when the PD/PGM pulse is low; when it is high (as in the case of the program pulse) the current drawn by this pin will be 30mA.

$ICC1$ is the power supply current when PD/PGM is high and V_{pp} is at a nominal 5V, and represents 25% of the total maximum ICC current. As was discussed previously, the outputs are automatically placed in the high impedance state when the PD/PGM pin is raised to V_{IH} . $ICC2$ is the maximum power supply current required by a 2716 in read mode, and reaches this maximum of 500mW ($30\mu W/bit$) at maximum temperature.

Table II. 2716 D.C. and Operating Characteristics.

$T_A = 0^\circ\text{C}$ to 70°C , $V_{CC}^{[1,2]} = +5\text{V} \pm 5\%$, $V_{PP}^{[2]} = V_{CC} \pm 0.6\text{V}^{[3]}$

Symbol	Parameter	Limits			Unit	Conditions
		Min.	Typ. ^[4]	Max.		
I_{LI}	Input Load Current			10	μA	$V_{IN} = 5.25\text{V}$
I_{LO}	Output Leakage Current			10	μA	$V_{OUT} = 5.25\text{V}$
$I_{PP1}^{[2]}$	V_{PP} Current			5	mA	$V_{PP} = 5.85\text{V}$
$I_{CC1}^{[2]}$	V_{CC} Current (Standby)		10	25	mA	$\overline{CE} = V_{IH}$, $\overline{OE} = V_{IL}$
$I_{CC2}^{[2]}$	V_{CC} Current (Active)		57	100	mA	$\overline{CE} = \overline{OE} = V_{IL}$
V_{IL}	Input Low Voltage	-0.1		0.8	V	
V_{IH}	Input High Voltage	2.2		$V_{CC}+1$	V	
V_{OL}	Output Low Voltage			0.45	V	$I_{OL} = 2.1\text{mA}$
V_{OH}	Output High Voltage	2.4			V	$I_{OH} = -400\mu\text{A}$

- NOTES: 1. V_{CC} must be applied simultaneously or before V_{PP} and removed simultaneously or after V_{PP} .
2. V_{PP} may be connected directly to V_{CC} except during programming. The supply current would then be the sum of I_{CC} and I_{PP1} .
3. The tolerance of 0.6V allows the use of a driver circuit for switching the V_{PP} supply pin from V_{CC} in read to 25V for programming.
4. Typical values are for $T_A = 25^\circ\text{C}$ and nominal supply voltages.
5. This parameter is only sampled and is not 100% tested.
6. t_{ACC2} is referenced to PD/PGM or the addresses, whichever occurs last.

All inputs are TTL compatible, requiring a V_{IL} between -0.1 and 0.8V and a V_{IH} of 2.2V minimum. Care should be exercised in selecting address buffers to ensure that the minimum V_{IH} level is met by use of appropriate TTL circuit elements or pull-up resistors to V_{CC} .

The outputs are also TTL compatible, producing a V_{OL} of 0.45V maximum at 2.1mA and a V_{OH} of 2.4V with -400mA capability.

A.C. Characteristics

Figure 7, the read mode timing indicates the maximum or minimum timing for the various timing parameters. Particular attention should be paid to

t_{DF} , chip deselect to output float time. This parameter indicates that the output buffers of the 2716 are not guaranteed to reach the high impedance state until 100ns after \overline{CS} reaches V_{IH} . If another device takes control of the output node before the first device output is in the high impedance state, excessive I_{CC} current will be drawn. See the Applications Section for further discussion.

Power Down Mode

The 2716 is the first MOS EPROM to have a completely static power down mode. This mode is activated by raising the \overline{CE} input to a TTL high level, with $V_{PP} = 5\text{V}$.

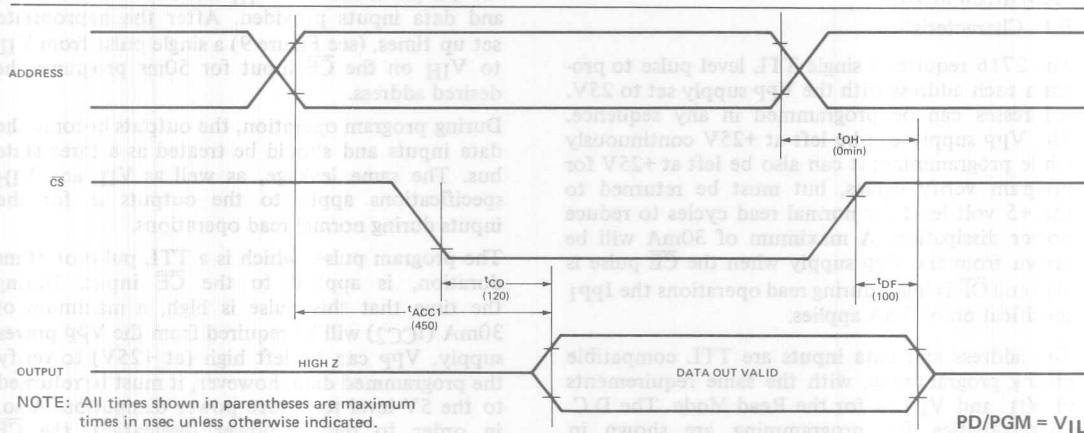
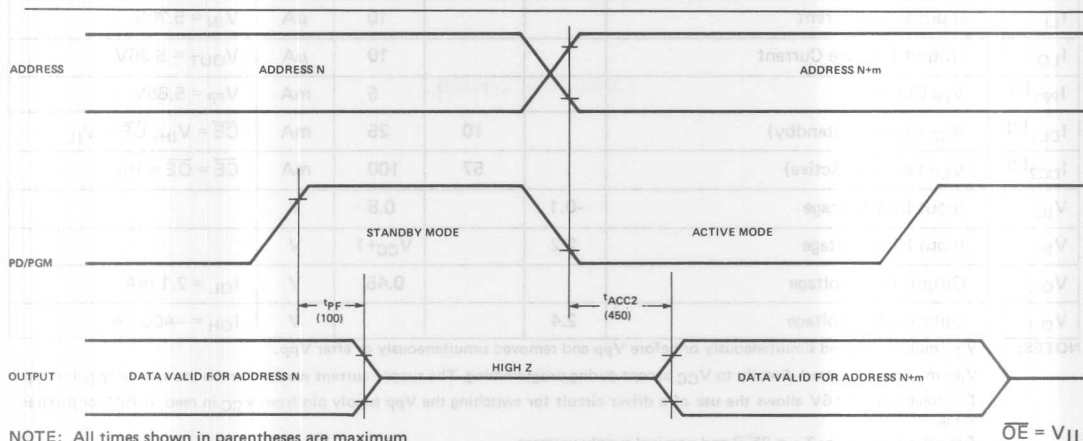


Figure 7. 2716 Read Waveforms.

The power is reduced by 75% (from 500mW to 125mW) during the time PD/PGM is high.

When the \overline{CE} pin is lowered to a TTL low level, the access time (t_{ACC2}) of 450ns is met as shown in

Figure 8. Of course, t_{ACC2} is referenced to either the addresses becoming stable or to the rising edge of \overline{CE} , whichever occurs last. Table III summarizes the A.C. Characteristics for both normal and power down read cycles.



NOTE: All times shown in parentheses are maximum times in nsec unless otherwise indicated.

Figure 8. 2716 Power Down Read Waveforms.

Table III. 2716 A.C. Characteristics.

$T_A = 0^\circ\text{C to } 70^\circ\text{C}$, $V_{CC}^{[1]} = +5V \pm 5\%$, $V_{PP}^{[2]} = V_{CC} \pm 0.6V^{[3]}$

Symbol	Parameter	Limits			Unit	Test Conditions
		Min.	Typ. ^[4]	Max.		
t_{ACC1}	Address to Output Delay		250	450	ns	$\overline{CE} = \overline{OE} = V_{IL}$
t_{ACC2}	PD/PGM to Output Delay		280	450	ns	$\overline{OE} = V_{IL}$
t_{CO}	Chip Select to Output Delay			120	ns	$\overline{CE} = V_{IL}$
t_{PF}	PD/PGM to Output Float	0		100	ns	$\overline{OE} = V_{IL}$
t_{DF}	Chip Deselect to Output Float	0		100	ns	$\overline{CE} = V_{IL}$
t_{OH}	Address to Output Hold	0			ns	$\overline{OE} = \overline{CE} = V_{IL}$

PROGRAM MODE

D.C. Characteristics

The 2716 requires a single TTL level pulse to program each address with the V_{PP} supply set to 25V. Addresses can be programmed in any sequence. The V_{PP} supply can be left at +25V continuously while programming; it can also be left at +25V for program verify cycles, but must be returned to the +5 volt level for normal read cycles to reduce power dissipation. A maximum of 30mA will be drawn from the V_{PP} supply when the \overline{CE} pulse is high and \overline{OE} is high; during read operations the I_{PP1} specification of 5mA applies.

The address and data inputs are TTL compatible during programming, with the same requirements of V_{IL} and V_{IH} as for the Read Mode. The D.C. Characteristics for programming are shown in Table IV. To enable the device for programming,

the \overline{OE} pin is taken to V_{IH} and the correct address and data inputs provided. After the appropriate set up times, (see Figure 9) a single pulse from V_{IL} to V_{IH} on the \overline{CE} input for 50ms programs the desired address.

During program operation, the outputs become the data inputs and should be treated as a three state bus. The same leakage, as well as V_{IL} and V_{IH} specifications apply to the outputs as for the inputs during normal read operations.

The program pulse, which is a TTL pulse of 50ms duration, is applied to the \overline{CE} input. During the time that this pulse is high, a maximum of 30mA (I_{CC2}) will be required from the V_{PP} power supply. V_{PP} can be left high (at +25V) to verify the programmed data, however, it must be returned to the 5V level to reduce power dissipation. Also, in order to reduce power dissipation, the \overline{CE} pulse must not be left high longer than 55ms when

Symbol	Parameter	Min.	Typ.	Max.	Units	Test Conditions
I_{LI}	Input Current (for Any Input)			10	μA	$V_{IN} = 5.25V/0.45$
I_{PP1}	V_{PP} Supply Current			5	mA	$\overline{CE} = V_{IL}$
I_{PP2}	V_{PP} Supply Current During Programming Pulse			30	mA	$\overline{CE} = V_{IH}$
I_{CC}	V_{CC} Supply Current			100	mA	
V_{IL}	Input Low Level	-0.1		0.8	V	
V_{IH}	Input High Level	2.2		$V_{CC}+1$	V	

- NOTES:**
1. Intel's standard product warranty applies only to devices programmed to specifications described herein.
 2. V_{CC} must be applied simultaneously or before V_{PP} and removed simultaneously or after V_{PP} . The 2716 must not be inserted into or removed from a board with V_{PP} at $25 \pm 1V$ to prevent damage to the device.
 3. The maximum allowable voltage which may be applied to the V_{PP} pin during programming is +26V. Care must be taken when switching the V_{PP} supply to prevent overshoot exceeding this 26V maximum specification.

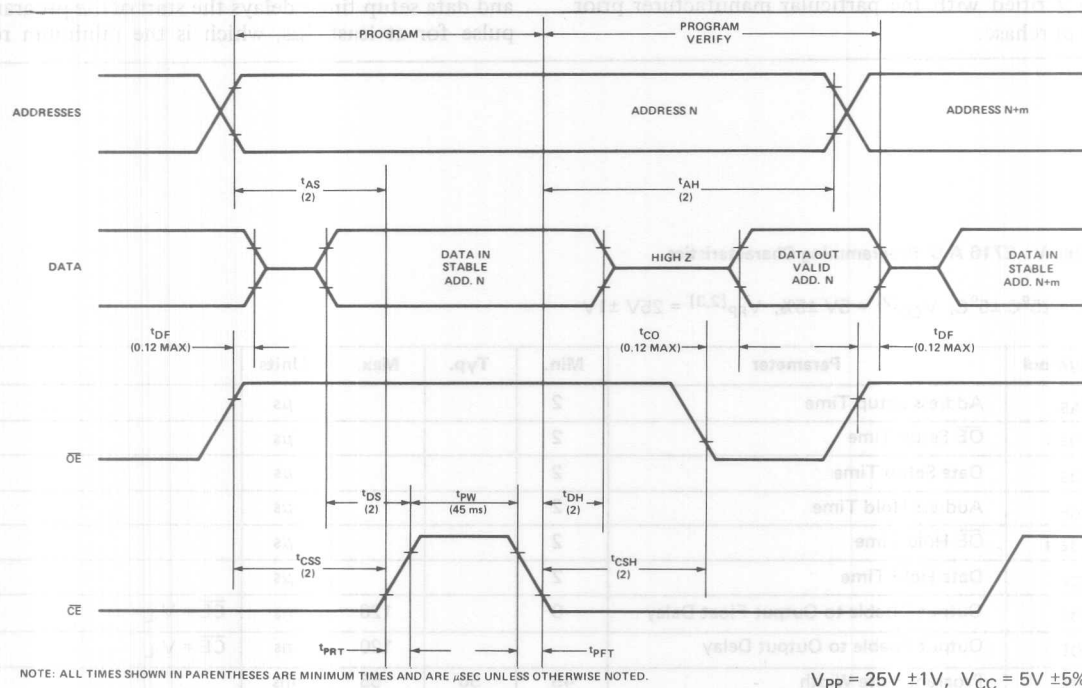


Figure 9. 2716 Programming Waveforms.

the V_{PP} supply is at +25V; it can be left high only with V_{PP} at +5V, which deselected the output and places the device in the low power standby mode.

The tolerance on the V_{PP} supply is $25V \pm 1V$. When switching the V_{PP} supply from +5V to +25V, particular care should be taken to ensure that there is no overshoot above 26V; exceeding this can be destructive to the programming circuits on the device. It is also not permitted to "hot socket" the device in a programmer (with respect to the V_{PP}

supply) as the resulting transients could cause the V_{PP} supply to exceed the maximum of 26V.

A.C. Characteristics

Figure 9 indicates the program mode timing, while Table V tabulates the various programming A.C. parameters.

To program a 2716, the address, data and \overline{CS} signals must all be stable $2\mu s$ before the PD/PGM pin is pulsed high for $50ms \pm 5ms$. This is shown in

Figure 9 as t_{AS} , t_{DS} and t_{CS} . After the falling edge of the program pulse, these same signals must be held stable for $2\mu s$ (t_{AH} , t_{DH} and t_{CSH}); then the next address and data can be presented, sequentially or not according to the ease of system implementation, and the next address programmed. In this manner it is possible to program an entire 2716 in approximately 100 seconds, while a single address requires only 50ms to program.

PROGRAMMING

A number of programmers are commercially available that will properly program the 2716. Intel maintains a service whereby commercial programmer manufacturers obtain design approval prior to marketing their device, in order to assure compatibility with Intel specifications. This approval should be verified with the particular manufacturer prior to purchase.

For those users who want to build their own programmer, a design is included at the end of this section.

Figure 10 illustrates a typical 2716 programmer block diagram. The address & data inputs can come from a system bus, or from toggle or thumbwheel switches. If system inputs are used, the Address Input Buffer should be a latch to allow the system bus to be free during the 50ms program time per address. The Data Input/Output Buffer should be of the bi-directional type to allow both programming and data verification.

The start control activates the timing chain to generate the required address and data setup and hold times, as well as the program pulse.

The program timer latches the address and data inputs stable and raises CS to V_{IH} , while the address and data setup timer delays the start of the program pulse for at least $2\mu s$, which is the minimum re-

Table V. 2716 A.C. Programming Characteristics.

$$T_A = 25^\circ\text{C} \pm 5^\circ\text{C}, V_{CC}^{[2]} = 5V \pm 5\%, V_{PP}^{[2,3]} = 25V \pm 1V$$

Symbol	Parameter	Min.	Typ.	Max.	Units	
t_{AS}	Address Setup Time	2			μs	
t_{OES}	\overline{OE} Setup Time	2			μs	
t_{DS}	Data Setup Time	2			μs	
t_{AH}	Address Hold Time	2			μs	
t_{OEH}	\overline{OE} Hold Time	2			μs	
t_{DH}	Data Hold Time	2			μs	
t_{DF}	Output Disable to Output Float Delay	0		120	ns	$\overline{CE} = V_{IL}$
t_{OE}	Output Enable to Output Delay			120	ns	$\overline{CE} = V_{IL}$
t_{PW}	Program Pulse Width	45	50	55	ms	
t_{PRT}	Program Pulse Rise Time	5			ns	
t_{PFT}	Program Pulse Fall Time	5			ns	

NOTES: 1. Intel's standard product warranty applies only to devices programmed to specifications described herein.

2. V_{CC} must be applied simultaneously or before V_{PP} and removed simultaneously or after V_{PP} . The 2716 must not be inserted into or removed from a board with V_{PP} at $25 \pm 1V$ to prevent damage to the device.

3. The maximum allowable voltage which may be applied to the V_{PP} pin during programming is +26V. Care must be taken when switching the V_{PP} supply to prevent overshoot exceeding this 26V maximum specification.

quired address and data setup time (t_{AS} and t_{DS}). The program pulse timer is activated by the falling edge of the address and data setup timer, and generates the required 50ms program pulse. The falling edge of the program pulse activates the address and data hold timer, ($2\mu s$ minimum) and the falling edge of the data hold timer resets the program times, releasing the latch on the address and data in buffers, freeing the system for either a verify cycle or a program cycle on another address.

On board programming is also very easily implemented with the 2716, as the \overline{CE} pin functions as a program inhibit, i.e., if a given device has \overline{OE} high, $V_{pp} = 25V$, and \overline{CE} low, it will not be programmed. A system showing how on-board programming could be implemented is shown in Figure 11. In the figure, device #4 will have address IFFH programmed with F4H, while the contents of address IFF in devices #1, #2 and #3 will be unaffected.

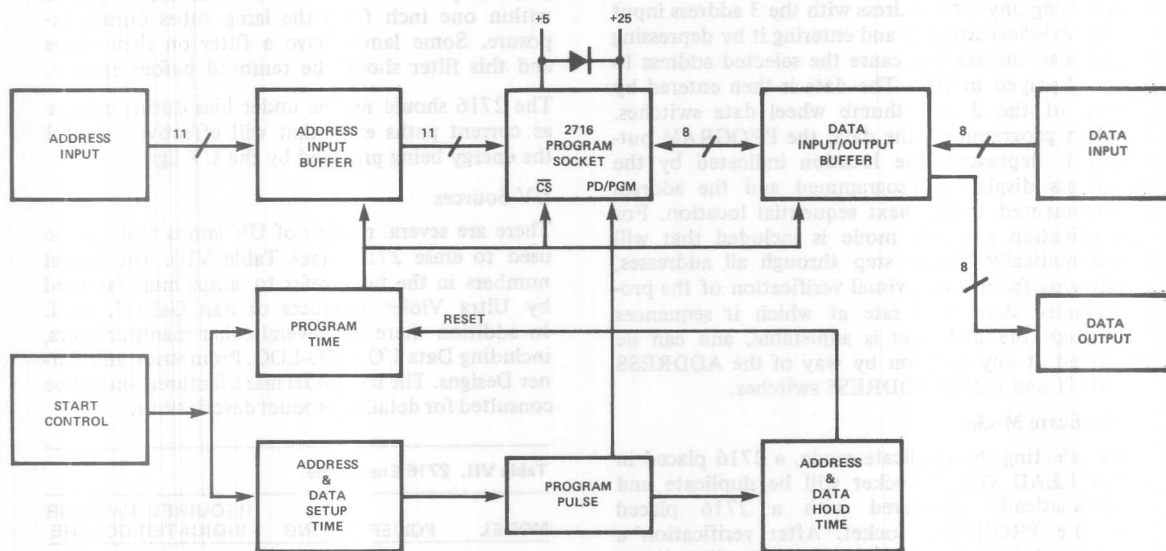


Figure 10. 2716 Programmer Block Diagram.

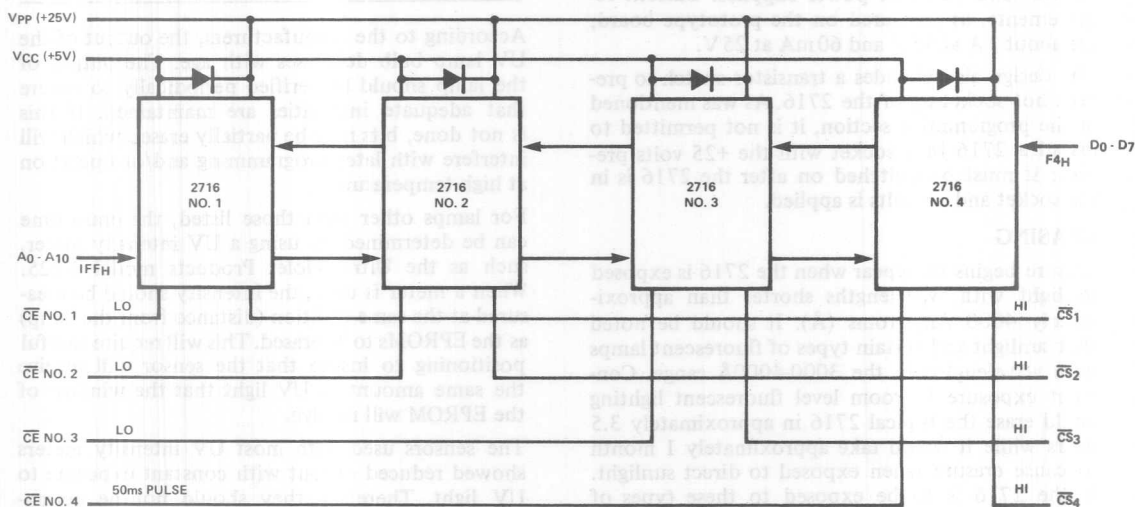


Figure 11. 2716 On Board Programming.

2716 Mini Programmer

Figure 12 presents the schematic for a 2716 programmer which is based on the block diagram shown in the previous section. This programmer has been design approved by Intel, by the same procedure used for commercial programmer manufacturers. The programmer has several features that make it useful for small development labs.

Manual Programming

Selecting any Hex address with the 3 address input thumb wheel switches and entering it by depressing the load button will cause the selected address to be displayed in Hex. The data is then entered by way of the 2 Hex thumb wheel data switches. When programming the data, the PROGRAM button is depressed, the location indicated by the address display is programmed and the address incremented to the next sequential location. For verification a verify mode is included that will automatically slowly step through all addresses, allowing for manual, visual verification of the programmed data. The rate at which it sequences through the addresses is adjustable, and can be started at any location by way of the ADDRESS INPUT and LOAD ADDRESS switches.

Duplicate Mode

By selecting the duplicate mode, a 2716 placed in the READ ONLY socket will be duplicate and automatically compared with a 2716 placed in the PROGRAM socket. After verification a green "PASS" or a red "FAIL" LED will indicate the completion of the program cycle. A blank check is not performed.

The design described here does not include a power supply design—the user must provide appropriate +5 volt and +25 volt power supplies. Current requirements, as measured on the prototype board, are about 1A at +5V and 60mA at 25V.

The design also includes a transistor switch to prevent hot socketing of the 2716. As was mentioned in the programming section, it is not permitted to install a 2716 in a socket with the +25 volts present: it must be switched on after the 2716 is in the socket and +5 volts is applied.

ERASING

Erase begins to appear when the 2716 is exposed to light with wavelengths shorter than approximately 4000 Angstroms (Å). It should be noted that sunlight and certain types of fluorescent lamps have wavelengths in the 3000-4000Å range. Constant exposure to room level fluorescent lighting could erase the typical 2716 in approximately 3.5 years while it would take approximately 1 month to cause erasure when exposed to direct sunlight. If the 2716 is to be exposed to these types of lighting conditions for extended periods of time, opaque labels are available from Intel which should

be placed over the 2716 window to prevent unintentional erasure.

The recommended erasure procedure for the 2716 is exposure to shortwave ultraviolet light which has a wavelength of 2537 Angstroms (Å). The integrated dose (i.e., UV intensity x exposure time) for erasure should be a minimum of 15 W-sec/cm². The erasure time with this dosage is approximately 20 minutes using an ultraviolet lamp with a 12000 μW/cm² power rating. The 2716 should be placed within one inch from the lamp tubes during exposure. Some lamps have a filter on their tubes and this filter should be removed before erasure.

The 2716 should not be under bias during erasure as current paths exist that will effectively cancel the energy being provided by the UV light.

UV Sources

There are several models of UV lamps that can be used to erase 2716's (see Table VII). The model numbers in the table refer to lamps manufactured by Ultra Violet Products of San Gabriel, Calif. In addition there are several other manufacturers, including Data I/O, PRO-LOG, Prometrics, and Turner Designs. The individual manufacturers should be consulted for detailed product descriptions.

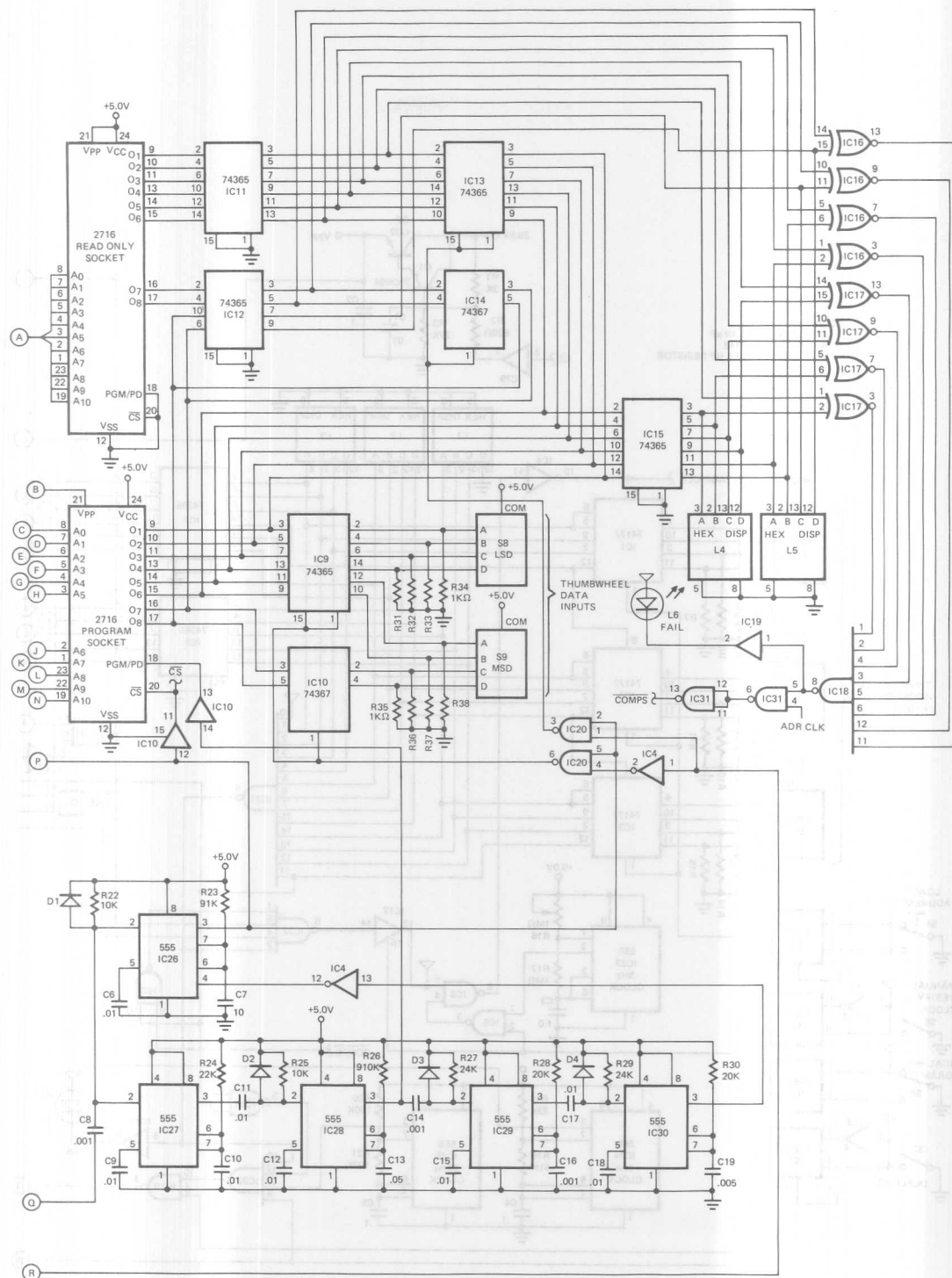
Table VII. 2716 Erase Time.

MODEL	POWER RATING	REQUIRED TIME FOR INDICATED DOSAGE
		15 W-sec 2716
R-52	13000μW/cm ²	19.2 min
S-52	12000μW/cm ²	20.7 min
S-68	12000μW/cm ²	20.7 min
UVS-54	5700μW/cm ²	43.8 min
UVS-11	5500μW/cm ²	45.6 min

According to the manufacturers, the output of the UV lamp bulb decreases with age. The output of the lamp should be verified periodically to ensure that adequate intensities are maintained. If this is not done, bits may be partially erased which will interfere with later programming and/or operation at high temperature.

For lamps other than those listed, the erase time can be determined by using a UV intensity meter, such as the Ultra Violet Products model J-225. When a meter is used, the intensity should be measured at the same position (distance from the lamp) as the EPROMs to be erased. This will require careful positioning to insure that the sensor will receive the same amount of UV light that the window of the EPROM will receive.

The sensors used with most UV intensity meters showed reduced output with constant exposure to UV light. Therefore they should not be permanently placed inside the erasure enclosure; they should only be used for periodic measurements.



Under Programming And Under Erasing

It is possible to "under program" the 2716 the same as it is with the 2708, such that the cell characteristic crosses the sense threshold. The result is that the cell apparently drops or picks up bits. As can be seen in Figure 13, the threshold characteristic has been shifted such that small changes in voltage or temperature will cause a "1" or a "0" to be sensed. This is always the result of insufficient erasing or programming. For programming to cause this problem, the device has only been partially programmed, and the characteristic curve has been shifted to the sense threshold point and the device will again seem to either pick up or drop bits. For erasure to cause the problem, the device has only been partially erased, such that the characteristic curve has only been shifted (right to left in the figure) to the threshold.

The cure in either case is to: 1) adequately erase by providing the required 15 W-sec/cm² of UV light at a frequency of 2537Å or; 2) program in accordance with the specifications.

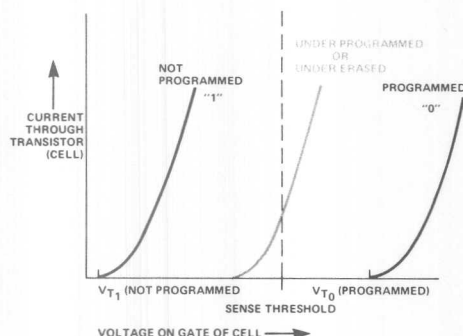


Figure 13. Effect of Under Programming or Under Erasure

2716 Mini Programmer

The Mini Programmer shown on the previous pages has been design approved by Intel and can be built as shown, or portions of the circuit can be modified to fit a specific user circuit application.

Circuit Description

The Mini Programmer has several modes of operation which are described below.

Manual Program — Controlled by pushbutton switch S6, this mode allows the user to program the address displayed by the address input displays (L1-L3) with the data that is entered in the data input thumbwheels (S8 & S9). The desired address to be programmed is entered by way of the LOAD ADDRESS switch, S4. This transfers the contents of the address input thumbwheel switches (S1-S3) to the address input buffers and the address display LEDs, L1-L3.

The desired data is entered in the form of two hexadecimal characters by way of the data input thumbwheel switches, S8 & S9. Prior to programming, the data output display will read FF_H, indicating that the addressed location contains all highs, i.e., is erased.

After the displayed address is programmed, the output display will momentarily display the contents of the programmed address, and then increment the address by 1 count, thus preparing the next sequential address to be programmed. Should other than the next sequential be desired, it is only necessary to dial in the new address and depress the LOAD ADDRESS pushbutton.

Manual Verify — In order to assure the user that the correct data pattern has been entered in an entire program, a manual verify function has been included. In this mode, the address counter will slowly cycle through addresses starting with the address that was loaded by the LOAD ADDRESS switch. The rate at which the counter will cycle is controlled by R16, and should be set for convenient visual recognition of the programmed data.

Duplicate Mode — Duplicate mode allows the contents of another 2716 to be programmed into an erased device that is inserted in the program socket. Each location is programmed and verified, and the next sequential location is programmed. Upon completion, PASS-FAIL indication is provided by way of LEDs L6 and L7.

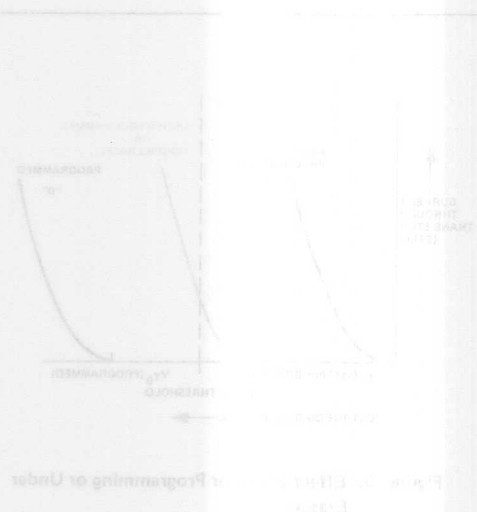
Transistors Q1 and Q2 provide for switching V_{pp} between 26V and 5V, while assuring that proper sequence and overshoot control is maintained.

Table VIII. 2716 Mini Programmer Parts List.

IC1-3	74177	4-Bit Counter
IC4	7404	Hex Driver
IC5	74279	Quad Set/Reset Latch
IC6, 20, 31	7400	Quad NAND
IC7-15	74367	Hex Tristate Driver
IC16, 17	74135	Quad Exclusive OR/NOR Gates
IC18	7430	8-Input NAND
IC19	7407	Open Collector, High Voltage Driver
IC21	74133	13-Input NAND
IC22	7420	Dual 4-Input NAND
IC23-30	NE555	Timer
Q1	MPS UO2	Transistor
Q2	2N3904	Transistor
R1	3K Ω	$\frac{1}{4}$ W Resistor
R2	820 Ω	$\frac{1}{4}$ W Resistor
R3	27K Ω	$\frac{1}{4}$ W Resistor
R4-15, 31-38	1K Ω	$\frac{1}{4}$ W Resistor
R16	1M Ω	Potentiometer (VERIFY Clock Rate)
R4-15	1K Ω	$\frac{1}{4}$ W Resistor
R31-38	1K Ω	$\frac{1}{4}$ W Resistor
R16	1M Ω	Potentiometer
R17	1M Ω	$\frac{1}{4}$ W Resistor
R18	33K Ω	$\frac{1}{4}$ W Resistor
R19	51K Ω	$\frac{1}{4}$ W Resistor
R20	750K Ω	$\frac{1}{4}$ W Resistor
R21	100K Ω	$\frac{1}{4}$ W Resistor
R22	10K Ω	$\frac{1}{4}$ W Resistor
R23	91K Ω	$\frac{1}{4}$ W Resistor
R24	22K Ω	$\frac{1}{4}$ W Resistor
R25	10K Ω	$\frac{1}{4}$ W Resistor
R26	910K Ω	$\frac{1}{4}$ W Resistor
R27, 29	24K Ω	$\frac{1}{4}$ W Resistor
R28, 30	20K Ω	$\frac{1}{4}$ W Resistor
C1, 6, 9-12, 15, 17, 18	0.01 μ F	Capacitor 20 wvdc (min)
C2, 4, 5	0.1 μ F	Capacitor 20 wvdc (min)
C3	1.0 μ F	Capacitor 20 wvdc (min)
C7	10 μ F	Capacitor 20 wvdc (min)
C8, 14, 16	0.001 μ F	Capacitor 20 wvdc (min)
C13	0.05 μ F	Capacitor 20 wvdc (min)
C19	0.005 μ F	Capacitor 20 wvdc (min)
S1-S3	(LSD-MSD): Address Input Switches (Cherry T-10 Thumbwheel)	
S4	Address Load (Pushbutton)	
S5	1Hz Verify Clock SPST Switch	
S6	Program Button (Pushbutton)	
S7	Duplicate Mode SPST Switch	
S8, S9	(LSD-MSD): Data Input (Cherry T-10 Thumbwheel)	
PROM Sockets	Textool 24-Pin ZIP DIP	
L1-L5	TIL311 Hexadecimal Display	
L6	MV5025 (Red LED)	
L7	MV5253 (Green LED)	

Under "Verify" mode, the 2716 the name of the program. The result is that the cell charge drops or picks up data. As can be seen in Fig. 13, the threshold characteristic has been shifted such that small changes in voltage or temperature will cause a "1" or a "0" to be read. This is the result of memory retention. For programming, the device has only been partially programmed. The characteristic curve has been shifted to either pick up or drop data. This causes the problem, the device is not fully erased, such that the characteristic curve has been shifted (right to left) to the threshold.

The next step is to (1) adequately erase by giving the device 15 W-seconds of UV light and (2) program the device in accordance with the instructions.



AP-78 DESIGN WITH EPROMS FOR FUTURE FLEXIBILITY

today's fast paced world of microprocessor system design. We must be practical in our designs and have flexibility to accommodate larger memory densities and more useful peripherals. Such a flexible, practical approach will allow us to design-in the future today. Designing EPROM systems based on speculation of future events is indeed a risky business. Vendors introduce devices with different pinouts, varying power requirements and unique densities. A more prevalent problem is one of industry wide availability; how can one possibly implement a firm design based on precarious and unpredictable supplies? How are designers to foresee rapid advances in EPROM densities? And, what about microprocessor evolution? Why should we implement a long range design when answers to such questions are so unsure? The reasons are clear. A small additional effort now will save a complete redesign and reimplementation in the years ahead. In one particular area—EPROM memory systems—the tools are available now, today.

The following paragraphs seek to refocus present EPROM design concepts. Flexible and creative approaches that encourage straightforward system evolution will be discussed. Simple and complex decoding schemes for device selection will be detailed and various control approaches explained. Logical design configurations that permit natural density upgrades will be noted. Finally, basic calculations aimed at determining memory speed requirements are discussed. These general system concepts are intended to inform the reader of recent developments that provide greater flexibility and system understanding.

In an ideal sense a flexible design would allow the use of all possible pinouts, package sizes, control schemes, and power requirements. To accomplish such universality one would have to make available all voltage and system signals and jumper them in at each device location. Figure 1 details this jumpered implementation. Admittedly, such a design would be inefficient—nevertheless, it fulfills all of our desired goals. Several power supply voltages are available, the pinout extends addressing up to 512K bits, the package is a relatively modest 28 pins, both single and dual control schemes are available. In the remaining paragraphs an attempt will be made to preserve that flexibility while making the implementation a more practical one.

BUS CONTENTION

The most fundamental decision a designer has to face involves solving the bus contention problem. Bus contention can arise when multiple devices are connected to a common data bus. If chip selection is accomplished only through address decoding then timing incompatibilities can result. The crucial timing parameters are address decode time (t_{ACC}), time from active chip enable to data valid (t_{CE}), and device deselect time (t_{DF}). Basically, contention occurs when one device is being selected while another is undergoing deselection. The

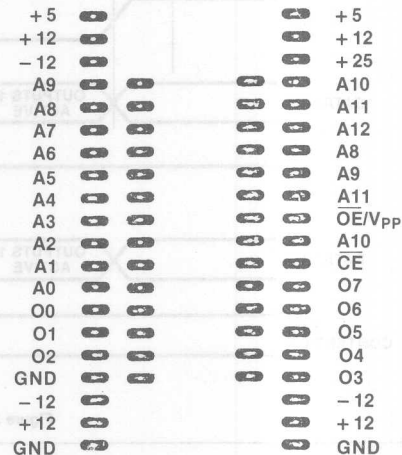


Figure 1.

worst case is when devices driving opposite logic levels are involved. A timing incompatibility results from fast decode times and relatively slow device turnoff times. Figure 2 illustrates the timing relationships while Figure 3 shows the physical circuit arrangement. The major problems that bus contention can cause are somewhat subtle in nature. Current and voltage spiking on the power supply rails is the most measurable one. Such noise can lead to a whole host of problems including invalid data, false triggering, race conditions, and reflections. In low performance systems these phenomena may have little effect—however—higher speed CPUs and mainframes can certainly be affected. Figure 4 shows a photograph of bus contention and its effects on circuit voltages and currents. Typically, the system designer solves the contention problem by making worst case timing calculations of decoder delays and EPROM turnoff times. Unfortunately, these parameters are subject to wide variation over time and temperature and correct designs on paper may not function in a realistic environment. When multiple cards with fast RAMs and PROMs are connected to a common bus the calculation and interaction becomes extremely complex. Schemes for device selection that rely only on decoded addresses (single line control) require intensive design efforts, with less than sure results. These schemes are thus prime candidates for bus contention problems.

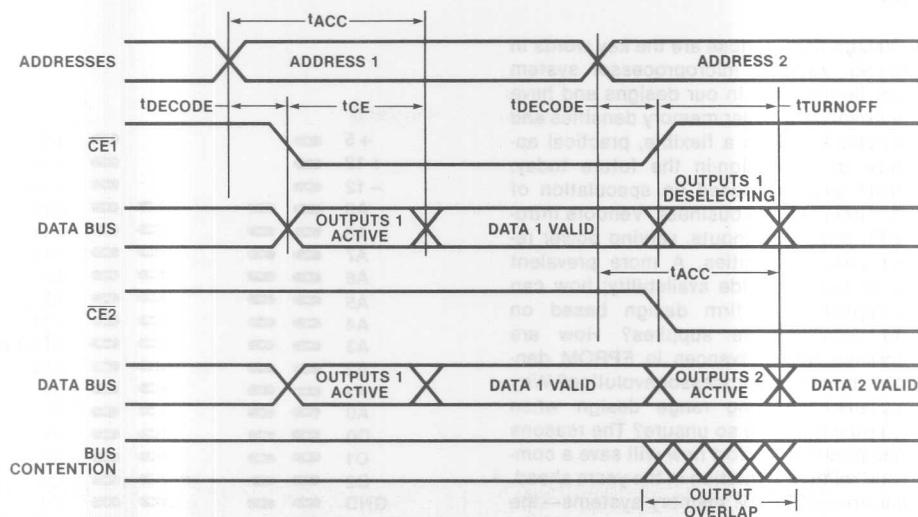


Figure 2. 1-Line Control Timing

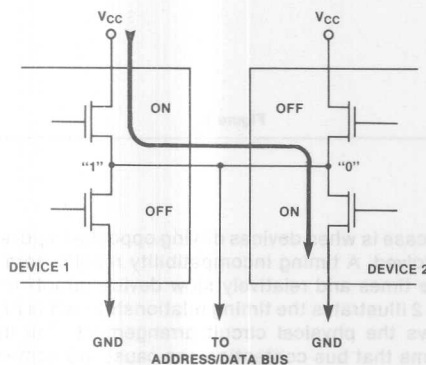


Figure 3. Bus Contention Path

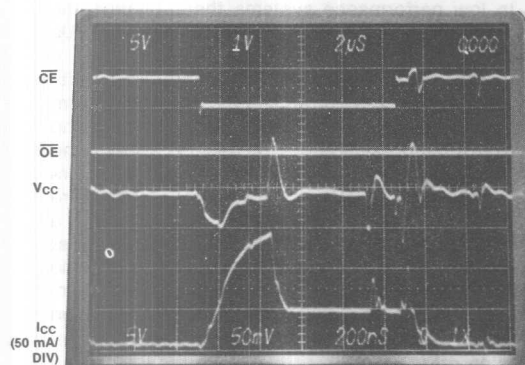


Figure 4.

Another means of device selection that completely eliminates the possibility of bus contention is a 2-line control scheme. All higher density Intel EPROMs possess this power down control architecture. Contention is eliminated through the use of an Output Enable (\overline{OE}) control pin. The \overline{OE} line controls the EPROM's tri-state output buffer and is designed to merge directly with the microprocessor outputs. The microprocessor allows data transfer through address decode and system control signals (typically \overline{RD} in Intel processors). Chip selection is based on address decode as before, however, the microprocessor controls when data is allowed onto the bus through the \overline{OE} pin. Generous timing between addresses and output enable guarantee that no contention will exist. Figure 5 illustrates 2-line control timing; Figure 6 shows oscilloscope photographs of system operation. It is clear that 2-line control frees the designer from much of the burden in assessing system timings. In addition, the design functionality is somewhat more assured than in a single control situation. Two-line control, an attribute of Intel products, thus reduces the burden that bus contention places on the system designer.

EPROM DENSITY UPGRADES

A fundamental advantage of flexible EPROM system design is the ability to increase storage capacity without hardware modification. The optimum situation would be one in which components of higher byte densities could be plugged directly into a socket used for lower densities. This gives the designer a wide range of options to tailor his memory system size to a particular application. A universal EPROM card is then available to a broad range of different products—saving design time and hardware costs. In designing such a system, several questions need to be addressed: Are EPROMs going to

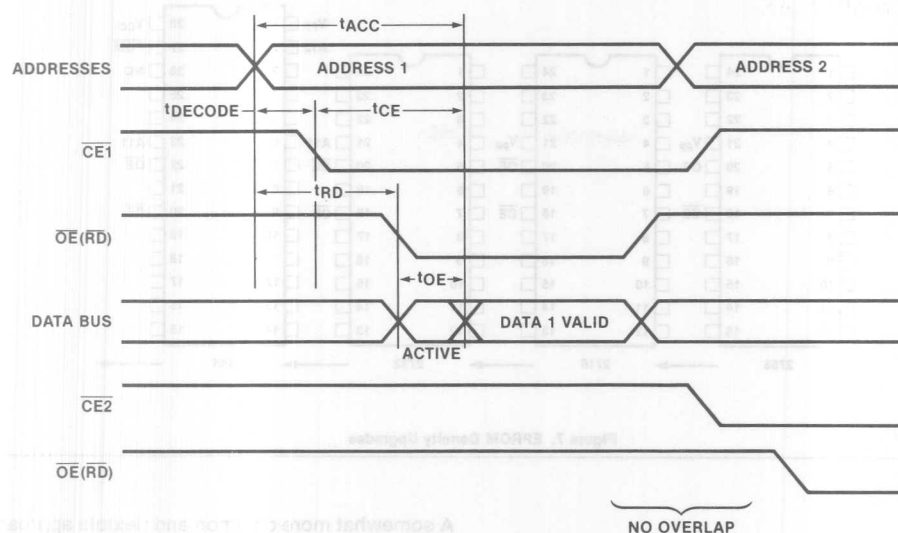


Figure 5. 2-Line Control Timing

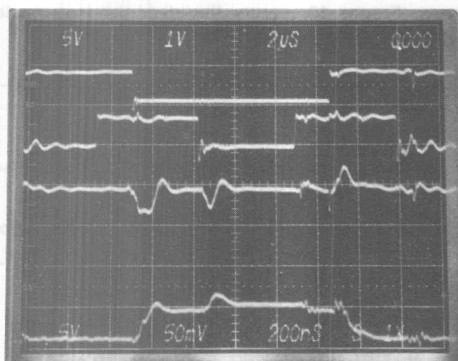


Figure 6.

be used for program store in a production mode? Are ROMs to be used to replace EPROMs during production? Is board space at a premium? Are second sources available?

Generally a present generation EPROM pinout determines the next generation ROM pinout. If one is planning to utilize ROMs in a production mode, this generality is significant. However, if EPROMs are to be used solely, one must consider the present EPROM pinout and foresee the changes that will allow higher densities. If many different pinouts become available simultaneously, we are offered a confusing choice in assessing what pinout will become a volume standard. Unfor-

tunately, we cannot accommodate all of the possible variations.

It is generally agreed that future developments in device density will occur through the conversion to 28 pin sites; such a direction is currently favored by the JEDEC standards committee. The advantage of such an approach is that lower capacity components can be inserted directly into the 28 pin site. This leaves the user with a wide range of board level densities to choose from. Figure 7 indicates one possible upgrade path for the near term; longer range possibilities are difficult to assess. In any event, the notable feature about the 28 pin site is its ability to accommodate very large EPROM densities. The addition of 4 pins allows a factor of 16 increase over the 2732 density. This future 512K bit device will even preserve the 2-line control scheme. Admittedly, these densities are some time away, however, more powerful microprocessors are becoming available that require large storage densities.

A basic design approach involves providing addresses on the board level for the highest foreseeable density (64K bytes in this case). Switches or jumpers then allow plug compatibility for different devices. One card accommodates all densities instead of a single board matched to a particular device. The flexibility gained would allow an "off the shelf" EPROM card that could be used for simple and complex system implementations. Product life can then be protracted because board level designs are kept intact for future developments. It would seem that the incremental cost in providing 28 pin sites is small in comparison to the benefit realized.

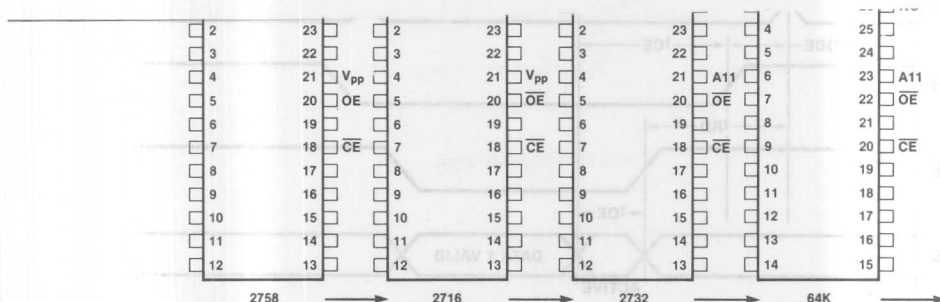


Figure 7. EPROM Density Upgrades

DECODING

Before any significant universality can be achieved, it is necessary to create a flexible decoding arrangement. Depending on the nature of the application, decoding schemes vary from the very simple to the extremely complex.

When there are few devices residing in system the most effective scheme is a simple NAND gate. Such an implementation is the lowest in cost, both from a component and power requirements standpoint. Figure 9 shows an arrangement for two devices. The major flaw with such an approach is the lack of simple density evolution. In addition, one must be careful to prevent bus contention when only using single line control (the rapid selection through the gate being the source of the problem).

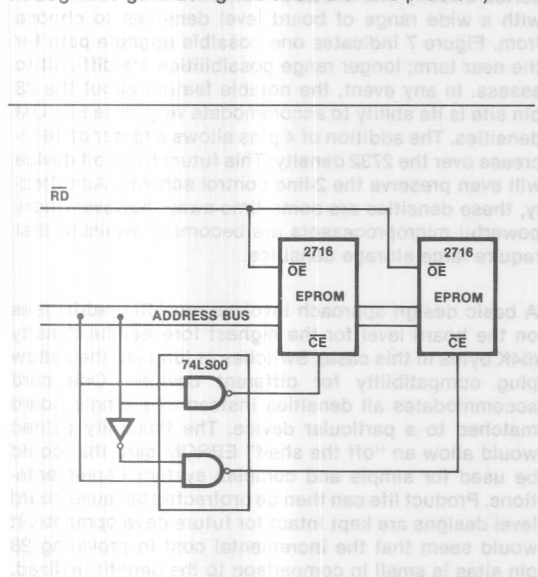


Figure 9. NAND Gate Decoding

A somewhat more common and flexible approach is the use of decoders such as the Intel 8205. Decoders allow selection of a greater number of devices yet, unfortunately, they lack the complexity to automatically compensate for changes in device density. Decoders are somewhat more expensive than single gate selection schemes. Figure 10 shows the decoder in a typical system application.

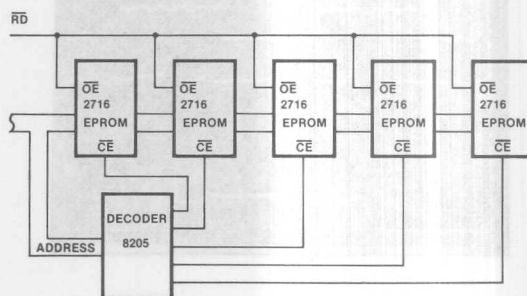


Figure 10. MSI Decoder Implementation

The most flexible means of addressing the device selection problem is an LSI approach. Dense Bipolar PROMs provide high speed as well as complex gating ability and thus offer a potent solution to direct density upgrades. The basic circuit operation allows the selection of different page boundaries, the PROM map then causes automatic segmentation. By programming the PROM with a universal decoding map, higher density parts can be inserted directly into 28 pin sites with no hardware modification. The PROM map is constructed in such a way as to maximize the number of different device densities that are plug compatible. In addition, the ability to interface 8 and 16 bit systems is also required. Figure 11

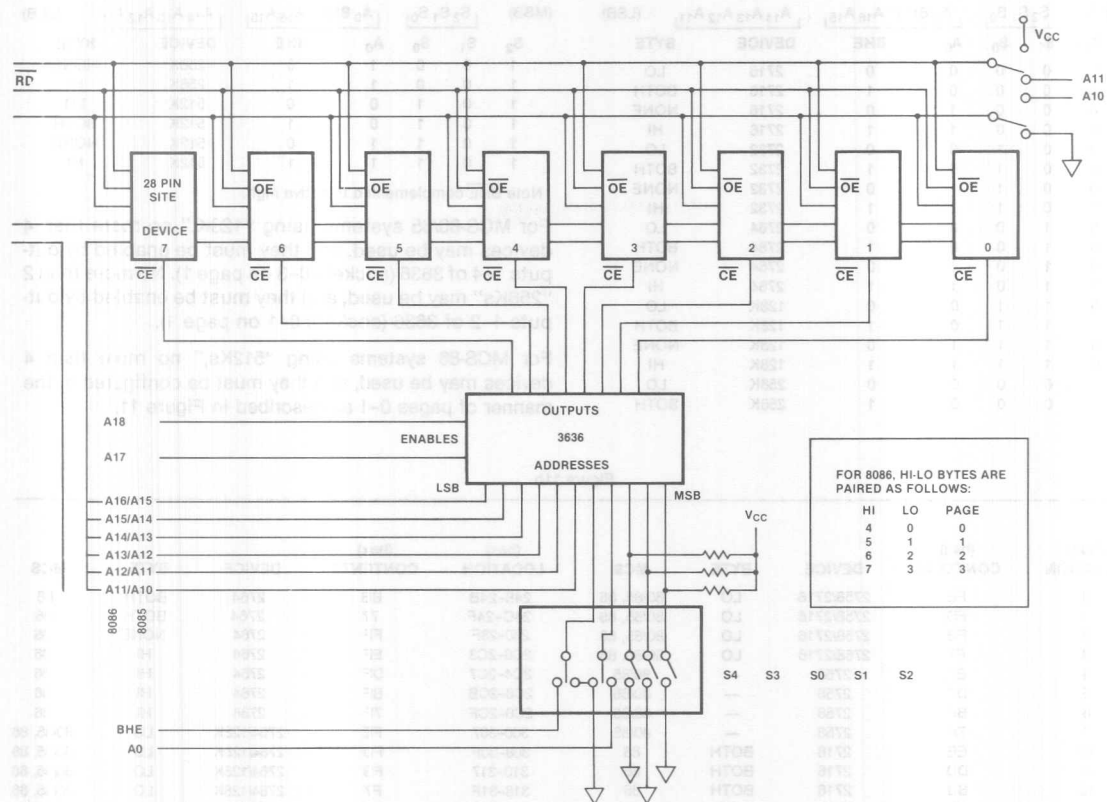


Figure 11. Universal PROM Decoder

For MCS-80/85, address input to 3636 looks like:

(MSB)	S ₂	S ₁	S ₀	0 0	A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	(LSB)
	S ₂	S ₁	S ₀								
	0	0	0								
	0	0	1								
	0	1	0								
	0	1	1								
	1	0	0								
	1	0	1								

Figure 11a.

shows the hardware implementation of such a scheme; Figure 12 is a PROM map that satisfies 8085, 8086, and 8088 processor interfaces with EPROM chip densities ranging from 1K byte to 64K bytes. In 8 bit micro-processor systems the decoder will handle 2716, 2732, 64K, 128K and 256K bit devices. The maximum decode capacity is 64K bytes. For 8086 systems, devices up to 512K bits can be decoded with a maximum capacity of 256K bytes.

In practice, the user sets the decoder page boundary by selecting switches S0-S2. The mode select table relates those switch positions to the device density that is to be used in the socket. Depending on the microprocessor status (8085, 8088, or 8086), switches S3 and S4 are adjusted. These switches allow the decoder to interpret byte or word accesses in 16 bit systems. As noted before, Figure 11 details such a scheme. Figures 11a and 11b indicate the switch positions and PROM addressing for 8085 and 8086 systems.

For MCS-86, address input to 3636 looks like:

(MSB)	S ₂	S ₁	S ₀	A ₀	BHE*	A ₁₆	A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	(LSB)	(MSB)	S ₂	S ₁	S ₀	A ₀	BHE*	A ₁₆	A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	(LSB)
	S ₂	S ₁	S ₀	A ₀	BHE	DEVICE	BYTE							S ₂	S ₁	S ₀	A ₀	BHE	DEVICE	BYTE					
	0	0	0	0	0	2716	LO							1	0	0	1	0	256K	NONE					
	0	0	0	0	1	2716	BOTH							1	0	0	1	1	256K	HI					
	0	0	0	1	0	2716	NONE							1	0	1	0	0	512K	LO					
	0	0	0	1	1	2716	HI							1	0	1	0	1	512K	BOTH					
	0	0	1	0	0	2732	LO							1	0	1	1	0	512K	NONE					
	0	0	1	0	1	2732	BOTH							1	0	1	1	1	512K	HI					
	0	0	1	1	0	2732	NONE																		
	0	0	1	1	1	2732	HI																		
	0	1	0	0	0	2764	LO																		
	0	1	0	0	1	2764	BOTH																		
	0	1	0	1	0	2764	NONE																		
	0	1	0	1	1	2764	HI																		
	0	1	1	0	0	128K	LO																		
	0	1	1	0	1	128K	BOTH																		
	0	1	1	1	0	128K	NONE																		
	0	1	1	1	1	128K	HI																		
	1	0	0	0	0	256K	LO																		
	1	0	0	0	1	256K	BOTH																		

*Note BHE complemented to active high.

For MCS-80/85 systems using "128K," no more than 4 devices may be used, and they must be enabled by outputs 1-4 of 3636 (sockets 0-3 on page 1). No more than 2 "256Ks" may be used, and they must be enabled by outputs 1-2 of 3636 (sockets 0-1 on page 1).

For MCS-86 systems using "512Ks," no more than 4 devices may be used, and they must be configured in the manner of pages 0-1 as described in Figure 11.

Figure 11b.

(hex)	(hex)				(hex)	(hex)			
LOCATION	CONTENTS	DEVICE	BYTE	MCS	LOCATION	CONTENTS	DEVICE	BYTE	MCS
0	FE	2758/2716	LO	80/85, 86	248-24B	BB	2764	BOTH	86
1	FD	2758/2716	LO	80/85, 86	24C-24F	77	2764	BOTH	86
2	FB	2758/2716	LO	80/85, 86	280-28F	FF	2764	NONE	86
3	F7	2758/2716	LO	80/85, 86	2C0-2C3	EF	2764	HI	86
4	EF	2758	—	80/85	2C4-2C7	DF	2764	HI	86
5	DF	2758	—	80/85	2C8-2CB	BF	2764	HI	86
6	BF	2758	—	80/85	2C6-2CF	7F	2764	HI	86
7	7F	2758	—	80/85	300-307	FE	2764/128K	LO	80/85, 86
40	EE	2716	BOTH	86	308-30F	FD	2764/128K	LO	80/85, 86
41	DD	2716	BOTH	86	310-317	FB	2764/128K	LO	80/85, 86
42	BB	2716	BOTH	86	318-31F	F7	2764/128K	LO	80/85, 86
43	77	2716	BOTH	86	320-327	EF	2764	—	80/85
80-83	FF	2716	NONE	86	328-32F	DF	2764	—	80/85
C0	EF	2716	HI	86	330-337	BF	2764	—	80/85
C1	DF	2716	HI	86	338-33F	7F	2764	—	80/85
C2	BF	2716	HI	86	340-347	EE	128K	BOTH	86
C3	7F	2716	HI	86	348-34F	DD	128K	BOTH	86
100-101	FE	2716/2732	LO	80/85, 86	350-357	BB	128K	BOTH	86
102-103	FD	2716/2732	LO	80/85, 86	358-35F	77	128K	BOTH	86
104-105	FB	2716/2732	LO	80/85, 86	380-39F	FF	128K	NONE	86
106-107	F7	2716/2732	LO	80/85, 86	3C0-3C7	EF	128K	HI	86
108-109	EF	2716	—	80/85	3C8-3CF	DF	128K	HI	86
10A-10B	DF	2716	—	80/85	3D0-3D7	BF	128K	HI	86
10C-10D	BF	2716	—	80/85	3D8-3DF	7F	128K	HI	86
10E-10F	7F	2716	—	80/85	400-40F	FE	128K/256K	LO	80/85, 86
140-141	EE	2732	BOTH	86	410-41F	FD	128K/256K	LO	80/85, 86
142-143	DD	2732	BOTH	86	420-42F	FB	128K/256K	LO	80/85, 86
144-145	BB	2732	BOTH	86	430-43F	F7	128K/256K	LO	80/85, 86
146-147	77	2732	BOTH	86	440-44F	EE	256K	BOTH	86
180-187	FF	2732	NONE	86	450-45F	DD	256K	BOTH	86
1C0-1C1	EF	2732	HI	86	460-46F	BB	256K	BOTH	86
1C2-1C3	DF	2732	HI	86	470-47F	77	256K	BOTH	86
1C4-1C5	BF	2732	HI	86	480-48F	FF	256K	NONE	86
1C6-1C7	7F	2732	HI	86	4CD-4CF	EF	256K	HI	86
200-203	FE	2732/2764	LO	80/85, 86	4D0-4DF	DF	256K	HI	86
204-207	FD	2732/2764	LO	80/85, 86	4E0-4EF	BF	256K	HI	86
208-20B	FB	2732/2764	LO	80/85, 86	4F0-4FF	7F	256K	HI	86
20C-20F	F7	2732/2764	LO	80/85, 86	500-51F	FE	256K/512K	LO	80/85, 86
210-213	EF	2732	—	80/85	520-53F	FD	256K/512K	LO	80/85, 86
214-217	DF	2732	—	80/85	540-55F	EE	512K	BOTH	86
218-21B	BF	2732	—	80/85	560-57F	DD	512K	BOTH	86
21C-21F	7F	2732	—	80/85	580-58F	FF	512K	NONE	86
240-243	EE	2764	BOTH	86	5C0-5DF	EF	512K	HI	86
244-247	DD	2764	BOTH	86	5E0-5FF	DF	512K	HI	86

Figure 12.

MEMORY PERFORMANCE CALCULATIONS AND THE MEMORY MATRIX

An extremely important factor in implementing a cost effective design is appropriate speed selection of EPROM devices. In determining required memory parameters, one must consider several factors. Processor speeds, card layout, buffer and latch delays, and capacitive loading effects all must be assessed. The following considerations are important in selecting the most appropriate and cost effective memory device.

An important decision regards the use of bus cycle wait states. If one can tolerate any number of wait state cycles, the following discussion will have little importance. However, if the decision is between none, 1, or 2 states, the memory matrix is a useful tool in making such a decision. The matrix (for Intel Microprocessors) is shown in Figure 13. It simply relates microprocessor type and system configuration to worst case memory speed requirements. The numbers in each of the matrix cells are worst case t_{CE} times. They represent the combination of all the system attributes on the matrix axis.

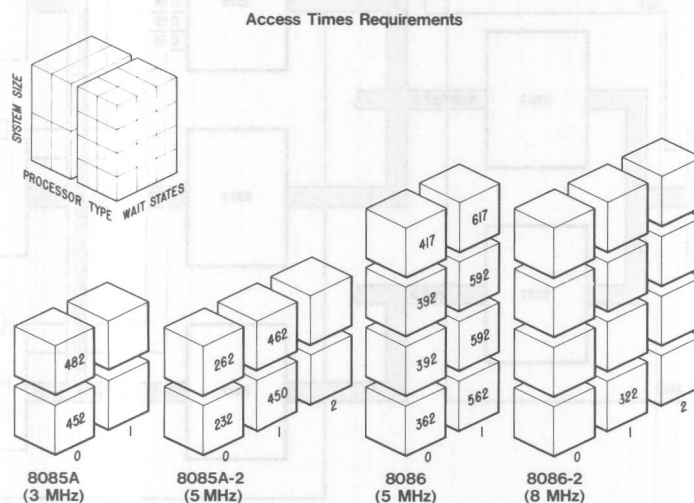
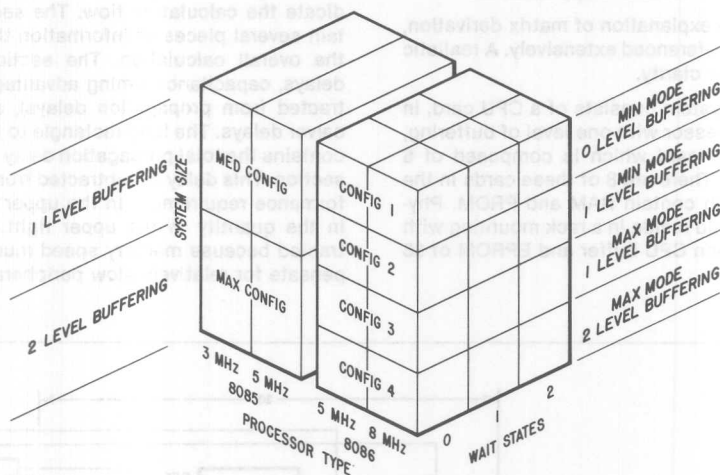


Figure 13. Microprocessor/Memory Matrix

To generate such a number array one must have a fairly complete understanding of the memory microprocessor interface. We will consider the system in Figure 14 and the relevant data paths noted there. The access scenario is as follows: The microprocessor issues an address onto its address/data bus. The multiplexed address is latched, and delayed; the address may be delayed further by bus drivers. The address is then decoded (another delay) and \overline{CE} reaches the memory device. Valid data then appears on the device output pins after t_{CE} . This data is propagated and delayed through levels of data transceivers and finally reaches the microprocessor. Depending on the circuit arrangement, wire lengths may increase these delays, and small capacitive loading may decrease them. The buffer and latch blocks may be several layers deep to accommodate different system sizes and applications.

To provide a complete explanation of matrix derivation, Figures 15 and 16 are referenced extensively. A realistic example is provided for clarity.

The microprocessor system consists of a CPU card, in this case an 8086 processor with one level of buffering, and a remote memory card which is composed of 6 EPROMs and a buffer. There are 8 of these cards in the system, some of which contain RAM and PROM. Physically, the system could reside in a rack mounting with a total distance between CPU buffer and EPROM of 36 inches.

Two calculations will be discussed—memory performance requirements and system timing margins. The memory performance parameters under consideration are t_{ACC} and t_{CE} . The output enable time t_{OE} can be calculated in a similar manner. We will proceed as follows:

1. Determine the processor requirements.
 - Time from addresses to data valid
2. Determine time from addresses to \overline{CE} .
 - Consider propagation delays
 - Consider capacitive effects
 - Consider wiring delays
3. Determine the propagation delay from memory device to microprocessor.

Referring now to Figure 15. Arrows within data paths indicate the calculation flow. The sectioned boxes contain several pieces of information that are pertinent to the overall calculation. The sections contain wiring delays, capacitance timing advantages (which are subtracted from propagation delays), and latch or transceiver delays. The long rectangle to the immediate right contains the total propagation delay through that circuit section. This delay is subtracted from the incoming performance requirement in the upper left box—resulting in the quantity in the upper right. The delay is subtracted because memory speed must increase to compensate for relatively slow peripherals. The calculation

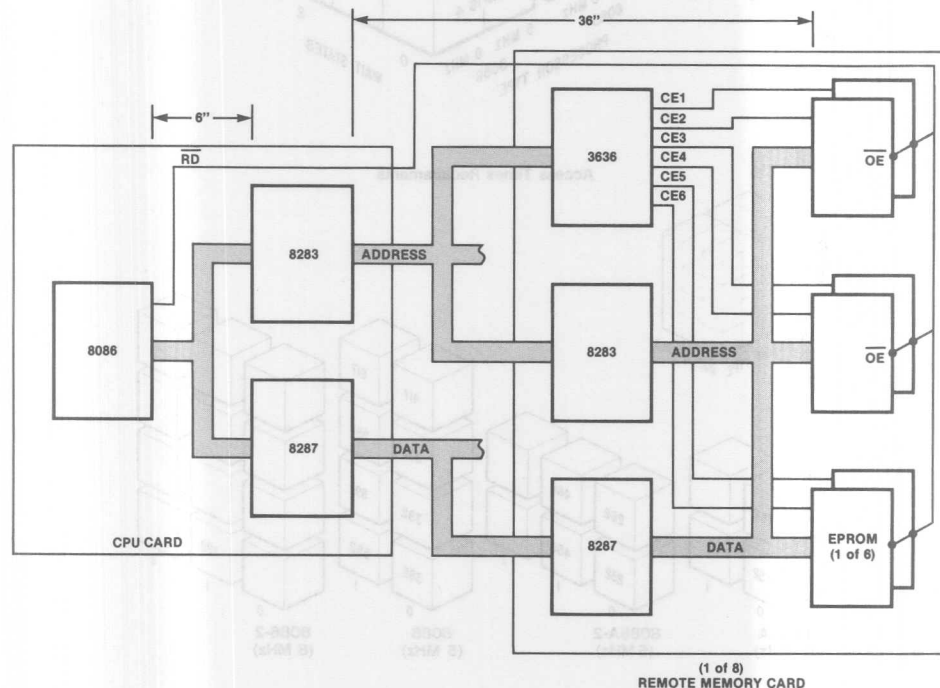


Figure 14. Timing Example

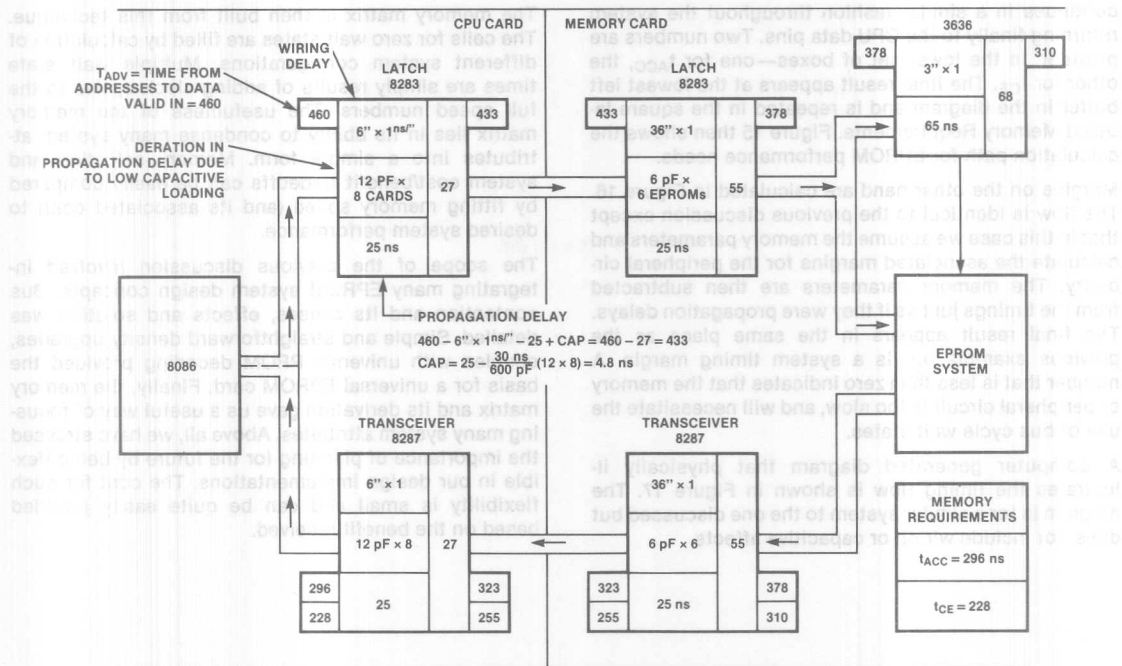


Figure 15. Memory Requirements Timing Flow

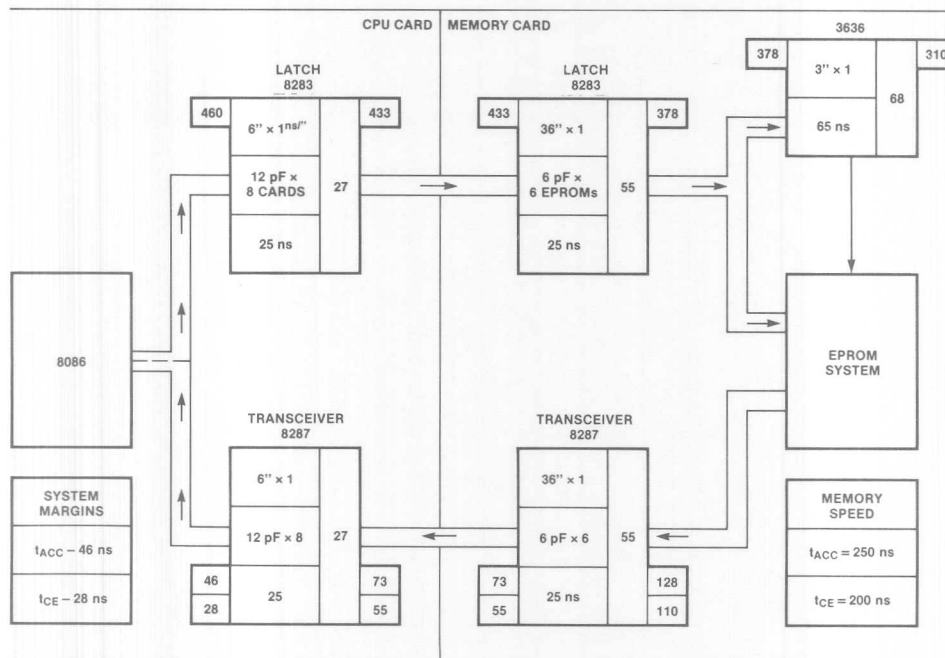


Figure 16. System Margins Timing Flow

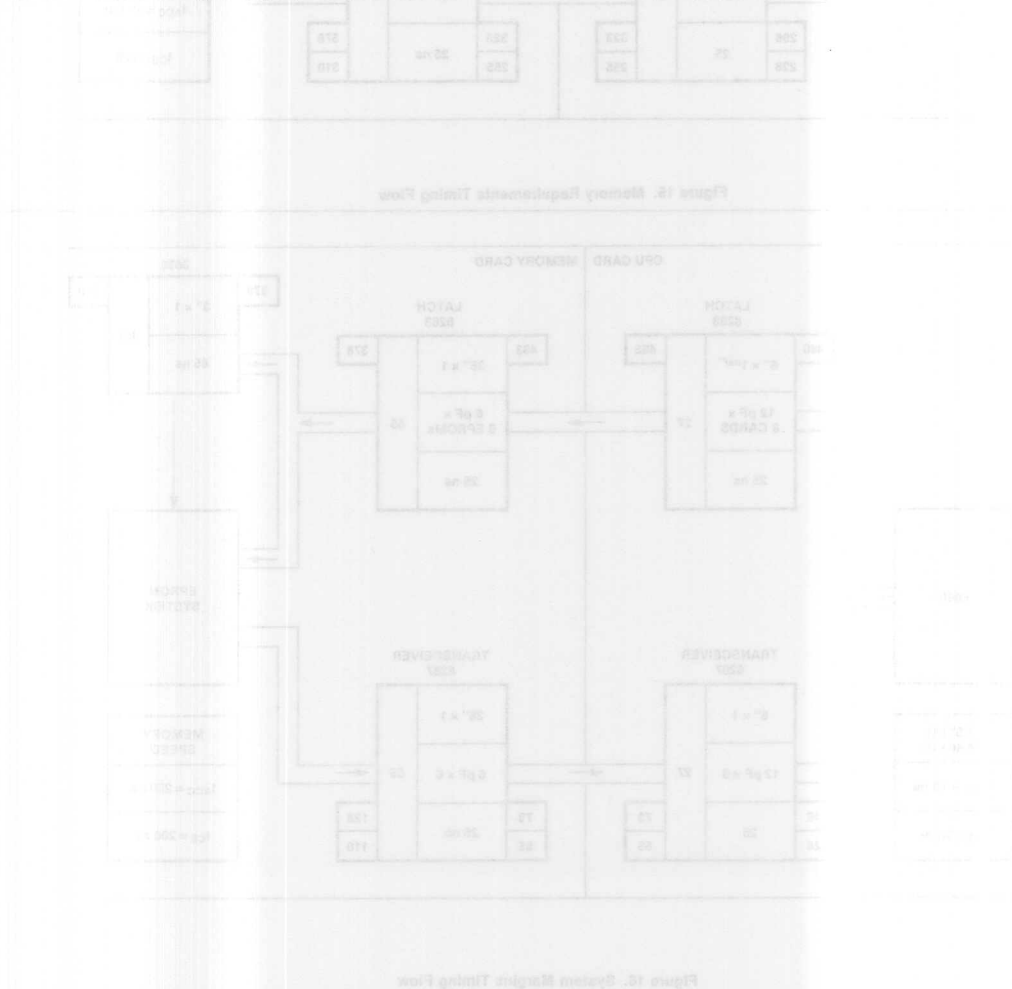
continues in a similar fashion throughout the system returning finally to the CPU data pins. Two numbers are provided in the lower set of boxes—one for t_{ACC} , the other for t_{CE} . The final result appears at the lowest left buffer in the diagram and is repeated in the square labeled Memory Requirements. Figure 15 then shows the calculation path for EPROM performance needs.

Margins on the other hand are calculated in Figure 16. The flow is identical to the previous discussion except that in this case we assume the memory parameters and calculate the associated margins for the peripheral circuitry. The memory parameters are then subtracted from the timings just as if they were propagation delays. The final result appears in the same place as the previous example but is a system timing margin. A number that is less than zero indicates that the memory or peripheral circuit is too slow, and will necessitate the use of bus cycle wait states.

A computer generated diagram that physically illustrates the timing flow is shown in Figure 17. The diagram is for a similar system to the one discussed but does not include wiring or capacitive affects.

The memory matrix is then built from this technique. The cells for zero wait states are filled by calculation of different system configurations. Multiple wait state times are simply results of adding clock cycles to the full speed numbers. The usefulness of the memory matrix lies in its ability to condense many system attributes into a simple form. Memory selection and system cost/benefit tradeoffs can be easily compared by fitting memory speed (and its associated cost) to desired system performance.

The scope of the previous discussion involved integrating many EPROM system design concepts. Bus contention and its causes, effects and solution was detailed. Simple and straightforward density upgrades, coupled with universal PROM decoding provided the basis for a universal EPROM card. Finally, the memory matrix and its derivation gave us a useful way of focusing many system attributes. Above all, we have stressed the importance of planning for the future by being flexible in our design implementations. The cost for such flexibility is small and can be quite easily justified based on the benefit received.



[illegible]

Step	Performance Level (Relative)
1	High
2	High
3	High
4	High
5	Low
6	Low-Mid
7	Low
8	Low-Mid
9	Low
10	Low-Mid

The diagram shows the timing of various signals for the 68000 microprocessor over 5 clock cycles. The signals and their active periods are as follows:

- STATUS**: Active (high) during the first two clock cycles.
- BHE**: Active (high) during the first two clock cycles.
- BHE/D**: Active (high) during the first two clock cycles.
- ALE**: Active (high) during the first two clock cycles.
- S2**: Active (high) during the first two clock cycles.
- AD0-15**: Active (high) during the first two clock cycles.
- AD0-15D**: Active (high) during the first two clock cycles.
- RD/**: Active (high) during the first two clock cycles.
- RD/D**: Active (high) during the first two clock cycles.
- MRDC/**: Active (high) during the first two clock cycles.
- MRDC/D**: Active (high) during the first two clock cycles.
- DT/R/**: Active (high) during the first two clock cycles.
- DEN/**: Active (high) during the first two clock cycles.
- CE/**: Active (high) during the first two clock cycles.
- OE/**: Active (high) during the first two clock cycles.
- DREQ**: Active (high) during the first two clock cycles.
- DCE/**: Active (high) during the first two clock cycles.
- DOE/**: Active (high) during the first two clock cycles.
- DBUS**: Active (high) during the first two clock cycles.

THE SCALE FACTOR IS: 100 ns

Figure 17. 8086 Configuration 4, 3625 Decode

INTRODUCTION — SYSTEM SPEED AND EFFICIENCY

With the advent of high performance microprocessing units, one must seriously consider the implications of processing speed and memory interface. As the processor becomes more and more powerful, with a higher throughput capability, the memory element in the system is burdened more heavily. The sophisticated and high performance 16-bit microprocessors (8086-2, the Z8000, the MC68000) require high performance memories. We are specifically referring to program store memories, PROMs or EPROMs. Because the microprocessor spends a large proportion of its time executing out of a program store medium, it is important that the memory be synergistically matched to the processing element. The throughput of the system as a whole depends on the speed at which the microprocessor cycles and the speed at which the memory elements can respond.

If the CPU can cycle very quickly and the memory elements cannot respond, the overall efficiency is limited by the memory access speeds. Thus, with high speed CPU and a low performance memory, the ability of the system to assimilate a large amount of information is limited. The system is incompatible and does not work in tandem. It is very important then to understand the interface between the microprocessing element and the non-volatile program store in order to realize efficient interaction.

The purpose of this discussion is to elaborate concepts regarding high performance CPUs and their effects on memory elements. We will consider the mathematical basis for the system as a whole; how the CPU, the peripheral delay circuitry, and the memory elements act in a predictable fashion. We will mathematically represent the system interface. Based on this mathematical representation we can practically apply the characteristics of external circuitry to the required memory access times. This can be related to the efficiency of the microprocessor with regard to operating cycle time and the use of wait states. We will also consider specifically the 8086, the 8085A-2, the Z8001 and the MC68000 with respect to their requirements for high performance non-volatile memories. The underlying goal is to address the concepts that will allow the system engineer to design a synergistically compatible system. The synergistic approach allows the use of high performance microprocessors in an environment where their ability can be fully realized. The use of the processor in a system with a large number of wait states is not a viable alternative. One does not gain any throughput or processing efficiency by using a low performance memory element with a high performance CPU. We will thus attempt to set forth concepts that are relevant to these high performance machines and demonstrate that with regard to future developments in memory access requirements, we have now reached a future with regard to access time. We will demonstrate that a sub-200 ns access memory

device is a requirement for high performance CPUs in medium to large system configurations.

THE EPROM MICROPROCESSOR INTERFACE

The evolution of the microprocessor has led us to a point where the ability of the device to address larger memory spaces has been increased dramatically. The first eight-bit microprocessing elements were able to address on the order of 64K bytes of information. However, with the new developments in the high performance microprocessor area, some devices can access up to 16 megabytes of information. As we have progressed to a point of requiring larger memory densities, microprocessor designers have had to deal with the trade-off between the pin density and system hardware requirements. There are two schools of thought in that area; one being the use of two separate buses for addressing information within the system. The 8080 is an example of such a structure; addresses are on one set of pins and data on another set. With the development of higher performance devices such as the 8085, we have seen a progression to a multiplexed bus architecture. This multiplexed architecture places both address and data on the same set of pins at different points in time. As we have moved to the high performance area a similar set of changes has occurred. The 8086 is an extension of the 8085 in that regard, as all sixteen address/data lines are multiplexed. There is an addition of four extra address lines for upward addressing capability, which are not multiplexed. The Zilog Z8001 is another device that uses a multiplexed bus architecture. The Motorola 68000 has taken the approach of keeping separate bus structures within the device; thus there are separate address and data pins. The disadvantage of the separate structure architecture is the large size of the package required due to the increased number of pins (40 for the 8086 vs. 48 for the Z8001 and 64 pins for the MC68000).

Because of the multiplexed architecture, requirements have been placed on the system as a whole to provide proper signals to the peripheral elements. If the memory and I/O elements within the system operate on a non-multiplexed basis, there is a requirement that the address/data bus be demultiplexed. In a large system configuration it would not be wise in any event to run a multiplexed bus structure over long distances without making proper level shifts and enhancements of noise level immunity. Thus in a medium to large system, there is a requirement that the signals coming out of the microprocessor be demultiplexed and their drive levels increased.

However, the addition of peripheral circuitry in interfacing to the memory elements in the system places a burden on those external elements. The microprocessor operates at a fixed rate, requesting information during a period of time that it requires. The external memory must respond within this time window. The time from

when the processor requests some action from the peripheral circuitry is fixed and finite. If this information does not arrive at the microprocessor during the correct time frame, then the device will execute based on the state of data bus. If the periphery does not respond the data bus will be indeterminate. Thus if there is any delaying circuitry between the microprocessor and the information source in the system, the requirement for speed is placed on the data source. The greater the delay between the microprocessor and the PROM element, the larger the burden on access speed.

For example: in a system that requires valid data 300 ns after addresses are sent out, we can discuss several situations. In the first case we can assume that there is no delay between the memory element and the microprocessing unit. In this instance a 300 ns access time is required. However, should we place latches and buffers between the microprocessor and the memory, perhaps with a delay of 50 ns, the requirement on the access time is 250 ns. Thus we see direct correlation between the amount of peripheral delay, the access speed of the memory, and the basic cycle time of the microprocessor. Since this memory cycle time is fixed, there is a greater requirement placed on the memory if the system is large.

Should we have a multcard system with latches and buffers, perhaps on each card level, we might experience at most three levels of delay in one direction to the memory element. This means that the addresses are propagated through three levels of gate delays, and the data returns through another three levels. Thus the total delay time through system can be as many as 6 levels. This is quite a significant burden to place on the memory elements. It indicates that as the microprocessing systems become higher performance with faster memory cycle times, requirements on access speeds increase. Three levels are said to be the maximum because of several considerations. First, we would expect that any signal leaving a card should be buffered, indicating one level of buffering. Second, we would expect to buffer signals as they enter a remote card. The third level would be located in a multiprocessing architecture with multiple processors on one card. Three levels are the very worst case.

There are several factors that impact the performance requirements of an EPROM device in the system. The first factor is the intrinsic characteristic of the CPU. That is, how does the rate at which the CPU runs affect the memory cycle time? For a fixed frequency, what equations determine the rate at which data is gained from the memory elements? The second consideration is the peripheral delay within the system: how much time is lost for latching, buffering, and driving of signals? The following requirement is an accumulation of the first two, that being the memory access time requirement. The memory requirements can be broken down into four factors. The first is the access time, t_{ACC} . The second, the time from chip enable to valid data, t_{CE} .

The third is the time from output enable to data valid, t_{OE} . The final consideration is the time from output enable inactive to data float, t_{DF} . The last two parameters mentioned refer to the output enable function which has been realized to eliminate bus contention within a system environment.

In summary, the memory access requirement is impacted directly by the speed at which the CPU operates and the amount of peripheral delay existing within the system. As the system grows larger, the peripheral delay increases, and the requirement on the memory becomes more stringent. The underlying concept is that the CPU should be able to operate at maximum speed without wait states. We can represent all of these variables in a fairly simple way. This is done through a concept called the surface of compatibility.

THE SURFACE OF COMPATIBILITY

The surface of compatibility is a mathematical concept that allows representation of the microprocessor and the EPROM interface. The concept is a universal one in that the microprocessor can operate at any frequency within its allowed range, and given any peripheral delay, the memory requirement can be calculated. Basically, the surface of compatibility is a result of a three-dimensional plot of the characteristics that we have just been discussing. If one plots the CPU clock frequency on one axis, and the peripheral delay in the system on the other axis, the resulting memory requirements can be represented on the final axis. If these characteristics are plotted over the entire range of the CPU operating frequency, a surface develops which indicates operation for zero wait states. Any point on the surface indicates a synergistic system interaction.

In plotting the characteristics, the CPU clock rate and the peripheral delay are the dependent variables. The memory access requirement is the independent variable and is calculated from the previous two factors. The clock rate is used to calculate the basic memory cycle requirement. Each microprocessor has a different characteristic equation governing the relationship between clock cycle time and memory cycle time. The equations for some common processors are listed in Figure 1. Typically, the relationship between clock cycles and

DEVICE	EQUATION	
8085A	$t_{adv} = (5/2 + N)T - 225 \text{ ns}$	MAXIMUM
8085A-2	$t_{adv} = (5/2 + N)T - 150 \text{ ns}$	MAXIMUM
8086	$t_{adv} = 3(TCLCL) - TCLAV - TDVCL$ $= (3 + N)T - 140 \text{ ns}$	MAXIMUM
8086-2	$t_{adv} = (3 + N)T - 80 \text{ ns}$	MAXIMUM
Z8001	$t_{adv} = (2 + N)T - 100 \text{ ns}$	MAXIMUM
MC68000	$t_{adv} = (3 + N)T - 150$ $t_{adv} = (3 + N)T - 100$	MAXIMUM TYPICAL

T = CLOCK PERIOD

Figure 1. Memory Cycle Time Equations

memory cycle time is a linear one; a certain number of clock cycles adjusted by some constant yields the memory cycle time.

The other variable, the peripheral delay associated with the buffering and latching circuitry is dependent on the physical characteristics of the system. If a certain kind of latch is used, it has associated with it a finite delay. This delay can be assumed to be the worst case for stringent engineering requirements, or in fact, can be a typical value depending on the system designer. With respect to the surface of compatibility, this peripheral delay includes only the address delay and the data delay between the memory and the microprocessor element. Thus if one has a latch and driver between the address pins of the CPU and the address pins of the memory, these devices represent the address delay. As the memory responds to those addresses, the delay in propagating to the CPU represents the data delay. Thus there are two levels of delay: address and data. Those accumulated are the total peripheral delay between the memory and the microprocessor. We can reduce the burden to the memory by reducing the delay through this peripheral circuitry.

The final independent variable, the memory access requirement, is calculated by taking the difference between the first two quantities mentioned. That is, if the microprocessor requires data 400 ns from addresses, and 200 ns of delay exists, the requirement on the memory is 200 ns. Thus we have a linear relationship between the memory cycle time and the peripheral delay. One simply subtracts the two to get the final memory access requirement.

The resulting three-dimensional plot of these factors yields the surface of compatibility for the system. Since the peripheral delay is plotted over a wide range, we can represent virtually any system configuration on this surface of compatibility. We do not have to tailor the calculation to one particular system. We can range the peripheral delay over many values, thus making it universal. For a large system we simply pick the associated peripheral delay and read the memory requirement off the access time axis. For example, given a fixed CPU clock cycle time of 4 megahertz we simply locate CPU speed against the associated peripheral delay of the system. Those two points locate a point on the surface of compatibility and indicate the memory access time requirement. Thus by describing the peripheral delay mathematically and the relationship between CPU clock cycle and memory access cycle mathematically, we can generate a universal plot of the performance of a CPU given any delay. The surface of compatibility shows us the relationship between these factors. It allows us to calculate the memory access requirement for any given CPU speed and any given peripheral delay.

The surface of compatibility itself may be somewhat academic, in that it only describes the system and does not provide us with an extremely convenient way of relating all these variables. The surface of compatibility being three dimensional is difficult to read and is difficult to obtain precise numbers from. However, the surface provides the basis for the generation of several things. It allows the generation of the memory matrix which is a concept that involves specific points on the surface of compatibility. It also allows the formulation of an access time requirements computer. This requirements computer is a practical implementation of the surface of compatibility.

The surface of compatibility is also useful from another respect. We have said that there is an intimate relationship between peripheral delay and memory access requirements. As the peripheral delay increases, the requirements on the memory become greater. On the other hand, if we have a fixed memory access speed, the associated timing margin within our system can be calculated. Thus there are two ways to look at the surface of compatibility. First, we can generate a memory performance requirement from the system parameters. Secondly, given the system parameters, the system margin can be determined.

To summarize, we have seen that the surface of compatibility is a way to conceptually represent the relationship between the CPU and the memory element over the operating range of the microprocessor. Any given CPU clock frequency and any peripheral delay will indicate one finite value for the memory access requirement. The surface of compatibility indicates this to us mathematically. The surface allows specific points to be calculated in the case of the memory matrix, and allows the generation of a memory access requirement calculator, which is a useful tool in addressing these concepts. The surface of compatibility relates the EPROM memory speed that the CPU demands on a system level.

SPECIFIC MEMORY REQUIREMENTS CALCULATIONS

Now that we have discussed the surface of compatibility as the basis for memory requirements calculations, they can be applied realistically to practical systems. Figure 2 is the surface of compatibility for the 8085A-2; a high performance 8-bit processor. Figures 3, 4 and 5 are surfaces of compatibility for high performance 16-bit microprocessors. There is one for the 8086 and 8086-2, one for the Z8001, as well as the MC68000. We will eventually discuss some specific system configurations which will indicate that all three of these microprocessors require high performance memories. These high performance memory requirements are access times on the order of 200 ns. The 8086 is somewhat less stringent than the other devices, yet all three require high performance memories. Such devices have not been

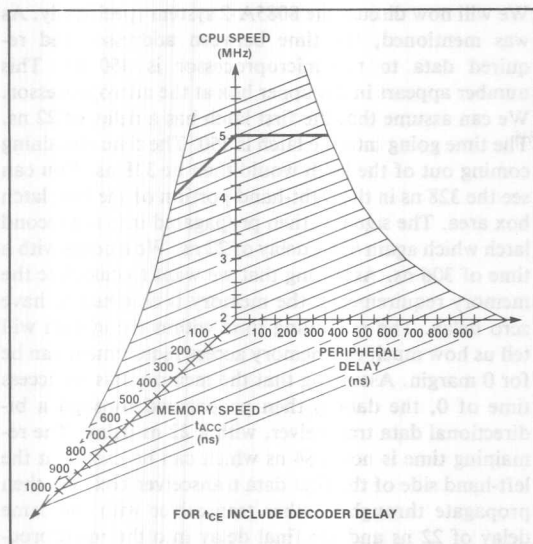


Figure 2. 8085A-2 Surface of Compatibility

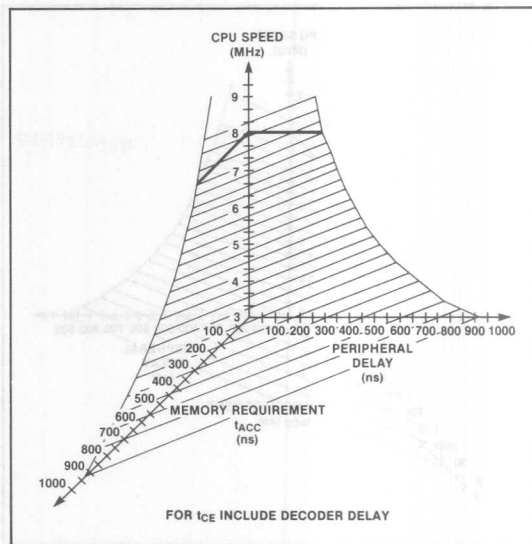


Figure 3B. 8086-2 Surface of Compatibility

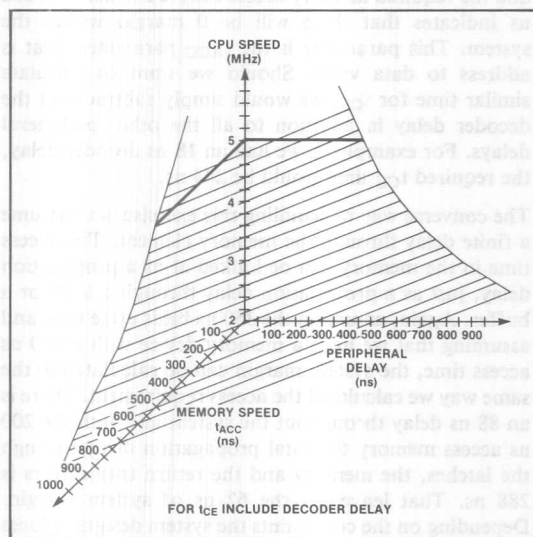


Figure 3A. 8086 Surface of Compatibility

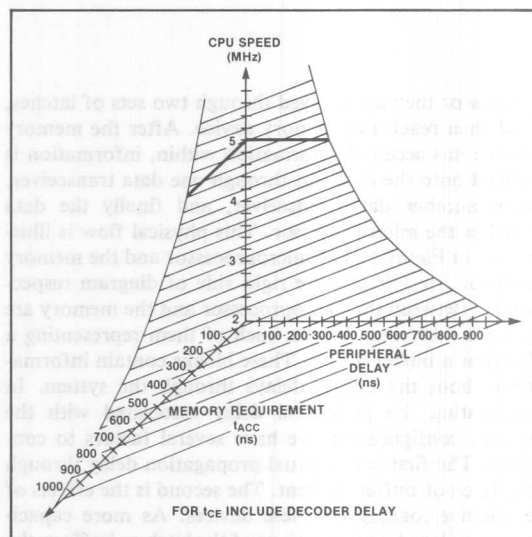


Figure 4. Z8000 Surface of Compatibility

available in the past. The new 2732A, recently introduced by Intel, is a 250 ns device with speed selection to 200 ns, in a $4K \times 8$ organization. Efficient system interaction is now preserved by providing high performance EPROM devices.

In calculating specific memory requirements there are two approaches. One approach involves using the surface of compatibility directly, the other involves diagramming the physical system with regard to timing. We will illustrate both methods and show that such theories are sound.

The first system configuration will be one that is based on the 8085A-2. The first step in using the surface of compatibility is to determine the operating frequency of the microprocessor. Assuming that the 8085 operates at its peak value of 5 megahertz, we can calculate through the equation in Figure 1 that the memory cycle time is 350 ns. The next factor to consider is the delay through address drivers, data buffers and data drivers. The system under consideration is assumed to have two sets of address buffer elements and two sets of data buffer elements. Thus, as the addresses emerge from the micro-

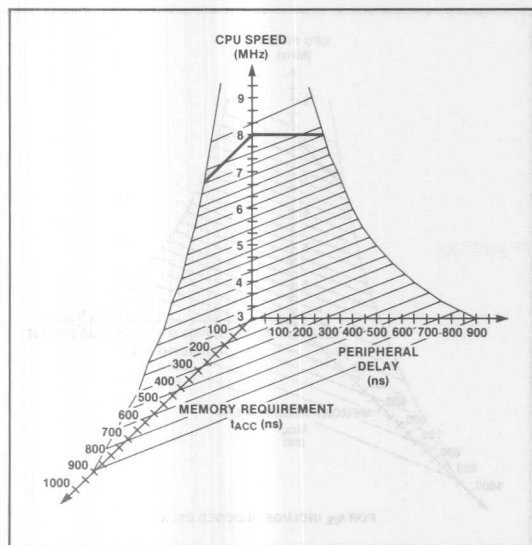


Figure 5. MC68000 Surface of Compatibility

processor they are delayed through two sets of latches, and then reach the memory device. After the memory device has accessed information within, information is placed onto the data bus through one data transceiver, then another data transceiver, and finally the data reaches the microprocessor. This physical flow is illustrated in Figure 6. The microprocessor and the memory exist on the left and the right side of diagram respectively. Between the microprocessor and the memory are the four different blocks, each of them representing a latch or a buffer device. These blocks contain information about the timing delays through the system. In calculating the peripheral delay associated with the system configuration, we have several factors to consider. The first is the actual propagation delay through the latch or buffer element. The second is the effects of capacitive loading on these devices. As more capacitance is placed on the outputs of the latch or buffers, the propagation delay can be expected to increase. The final factor is the length of wiring between the buffer and the next stage in the system. We would expect that as the distances between the buffering elements get greater, the peripheral delay increases as well. We can figure on the order of 1 ns per inch for wiring delay. For purposes of this discussion, we will simplify the argument by saying that the elements within the system will not be fully loaded capacitively and that the wiring delays are fairly short. We can assume that the contribution due to capacitance and the contribution due to wiring will offset one another; we will only consider the propagation delay through the latch or buffer device.

We will now discuss the 8085A-2 system specifically. As was mentioned, the time between addresses and required data to the microprocessor is 350 ns. This number appears in the upper box at the microprocessor. We can assume that the first latch has a delay of 22 ns. The time going into the latch is 350. The time remaining coming out of the latch would then be 328 ns. You can see the 328 ns in the right-hand portion of the first latch box area. The signal is then propagated into the second latch which again has a delay of 22 ns. We emerge with a time of 306 ns. Assuming that we want to calculate the memory requirements, the memory is assumed to have zero delay. The remaining time within the system will tell us how much the memory access requirement can be for 0 margin. Assuming that the memory has an access time of 0, the data is then propagated through a bi-directional data transceiver, with a 22 ns delay. The remaining time is now 284 ns which can be shown at the left-hand side of the first data transceiver box. We then propagate through another transceiver with the same delay of 22 ns and the final delay into the microprocessor is 262 ns. Thus we have seen that the peripheral delay associated with the system configuration is 88 ns and the required memory access time is 262 ns. This 262 ns indicates that there will be 0 margin within the system. This parameter is the t_{ACC} parameter, that is address to data valid. Should we want to calculate similar time for t_{CE} , we would simply subtract out the decoder delay in addition to all the other peripheral delays. For example, if we had an 18 ns decoder delay, the required t_{CE} time would be 244 ns.

The converse way of handling this exercise is to assume a finite delay through the memory element. The access time in the memory can be looked at as a propagation delay, just as a propagation delay through a latch or a buffer. Again, assuming the 350 ns basis cycle time and assuming that we have a memory device with a 200 ns access time, the system margin can be calculated in the same way we calculated the access requirement. There is an 88 ns delay throughout the system and with the 200 ns access memory the total propagation delay through the latches, the memory and the return transceivers is 288 ns. That leaves us the 62 ns of system margin. Depending on the constraints the system designer places on the timing margins, this may be adequate or too little.

A similar calculation can be done for the t_{OE} parameter, the time from when output enable is active to data required at the microprocessor. The critical time path is different because the relevant control signal is not an address line but a processor control line, the read line. The calculations would be as follows: the time from read active to data valid is 150 ns. Assuming that there is no buffering delay of the read signal, then read reaches the memory element directly from the microprocessor. Assuming again that the t_{OE} would be 0 to calculate the system margin we then start with 0 ns at the data

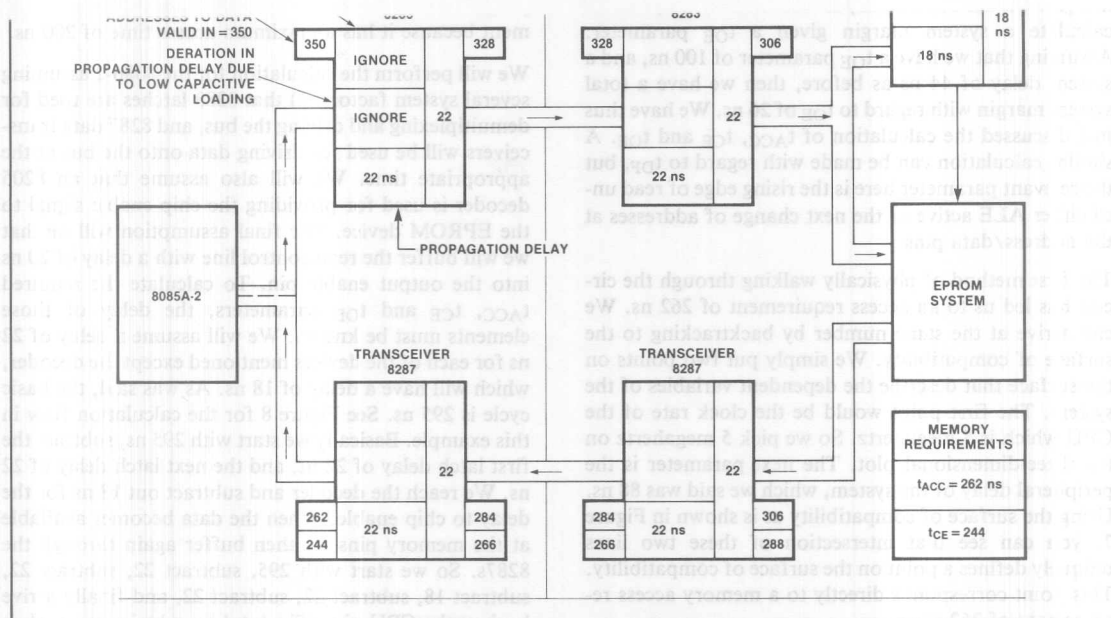


Figure 6A. Physical Timing Path

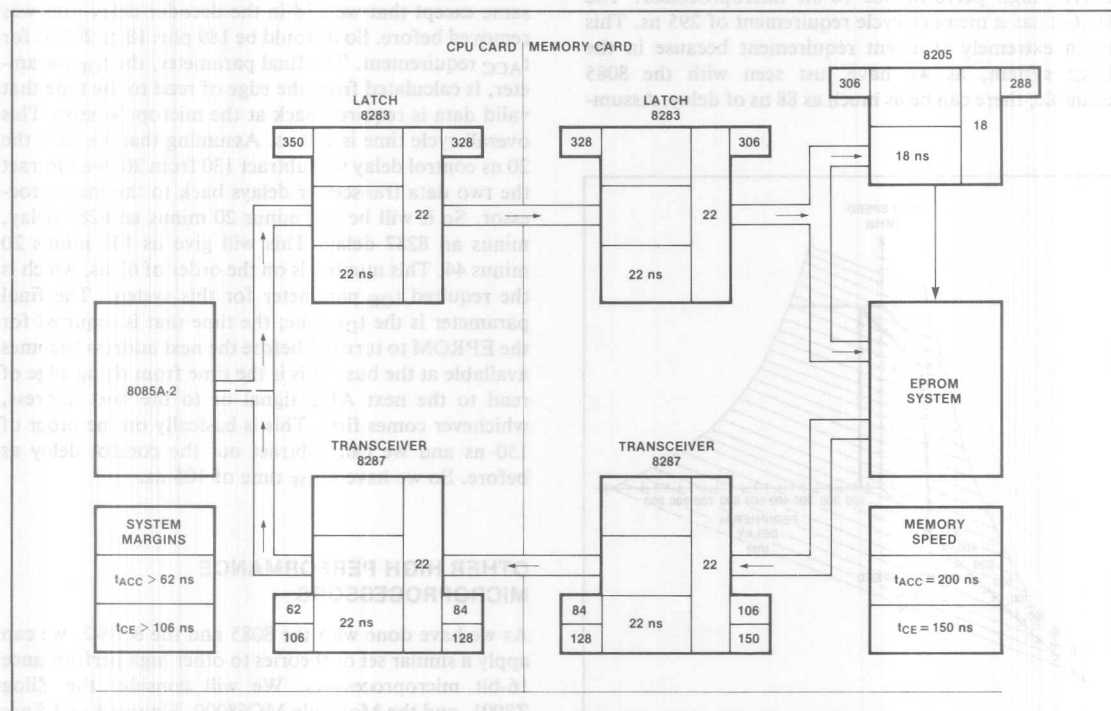


Figure 6B. Physical Timing Path

transceiver. The first data transceiver gives us a delay of 22 ns, the second another delay of 22 ns. Thus, the t_{OE} parameter is on the order of 106 ns assuming 0 system margin. On the other hand, as was done before, we can calculate a system margin given a t_{OE} parameter. Assuming that we have a t_{OE} parameter of 100 ns, and a system delay of 44 ns as before, then we have a total system margin with regard to t_{OE} of 26 ns. We have thus just discussed the calculation of t_{ACC} , t_{CE} and t_{OE} . A similar calculation can be made with regard to t_{DF} , but the relevant parameter here is the rising edge of read until either ALE active or the next change of addresses at the address/data pins.

The first method of physically walking through the circuit has led us to an access requirement of 262 ns. We can arrive at the same number by backtracking to the surface of compatibility. We simply put two points on the surface that describe the dependent variables of the system. The first point would be the clock rate of the CPU which is 5 megahertz. So we pick 5 megahertz on the three-dimensional plot. The next parameter is the peripheral delay of the system, which we said was 88 ns. Using the surface of compatibility as is shown in Figure 7, you can see that intersection of these two lines uniquely defines a point on the surface of compatibility. This point corresponds directly to a memory access requirement of 262 ns.

We can do a similar calculation for the 8086-2, which is a very high performance 16-bit microprocessor. The 8086-2 has a memory cycle requirement of 295 ns. This is an extremely stringent requirement because in the large system, as we have just seen with the 8085 example, there can be as much as 88 ns of delay. Assum-

ing the same 88 ns for the 8086-2 system, the required memory speed would be 207 ns. This demonstrates the need for a high performance EPROM for non-volatile program store. The 2732A-2 would satisfy this requirement because it has a maximum access time of 200 ns.

We will perform the calculation for the 8086-2 assuming several system factors: 1) that 8283 latches are used for demultiplexing and driving the bus, and 8287 data transceivers will be used for driving data onto the bus at the appropriate time. We will also assume that an 8205 decoder is used for providing the chip enable signal to the EPROM device. The final assumption will be that we will buffer the read control line with a delay of 20 ns into the output enable pin. To calculate the required t_{ACC} , t_{CE} and t_{OE} parameters, the delay of those elements must be known. We will assume a delay of 22 ns for each of the devices mentioned except the decoder, which will have a delay of 18 ns. As was said, the basic cycle is 295 ns. See Figure 8 for the calculation flow in this example. Basically we start with 295 ns, subtract the first latch delay of 22 ns, and the next latch delay of 22 ns. We reach the decoder and subtract out 18 ns for the delay to chip enable. When the data becomes available at the memory pins we then buffer again through the 8287s. So we start with 295, subtract 22, subtract 22, subtract 18, subtract 22, subtract 22, and finally arrive back at the CPU pins. The total round trip time is then 295 ns minus 88, minus 18. This gives us a requirement on the t_{CE} time of the memory of 189 ns. The t_{ACC} is the same except that we add in the decoder delay that was removed before. So it would be 189 plus 18 or 207 ns for t_{ACC} requirement. The final parameter, the t_{OE} parameter, is calculated from the edge of read to the time that valid data is required back at the microprocessor. This overall cycle time is 130 ns. Assuming that we have the 20 ns control delay we subtract 130 from 20, we subtract the two data transceiver delays back to the microprocessor. So it will be 130 minus 20 minus an 8287 delay, minus an 8287 delay. This will give us 130 minus 20 minus 44. This number is on the order of 62 ns, which is the required t_{OE} parameter for this system. The final parameter is the t_{DF} time; the time that is required for the EPROM to turn off before the next address becomes available at the bus. This is the time from rising edge of read to the next ALE signal or to the next address, whichever comes first. This is basically on the order of 150 ns and we can subtract out the control delay as before. So we have a t_{DF} time of 106 ns.

OTHER HIGH PERFORMANCE MICROPROCESSORS

As we have done with the 8085 and the 8086-2, we can apply a similar set of theories to other high performance 16-bit microprocessors. We will consider the Zilog Z8001, and the Motorola MC68000. Figures 4 and 5 are the surface of compatibility plots for these two devices.

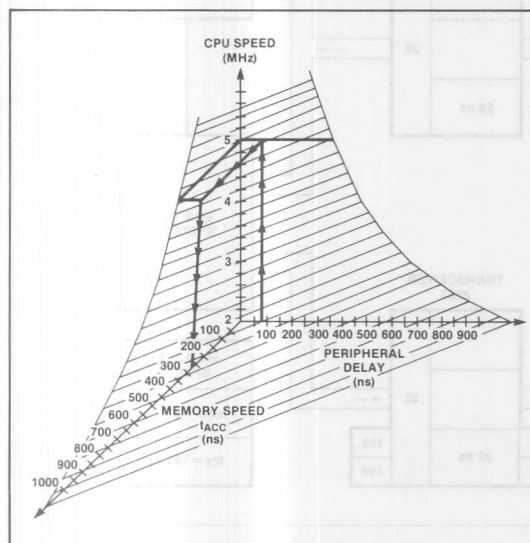


Figure 7. Surface of Compatibility for System

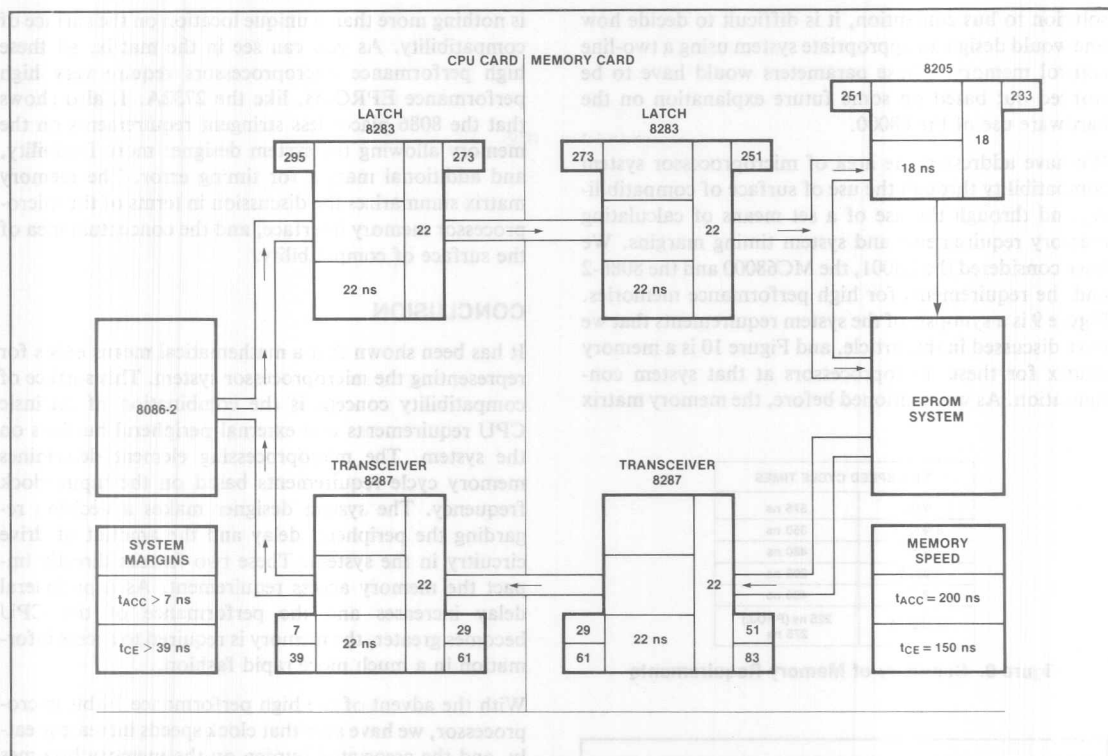


Figure 8. 8086-2 System Flow

We will first consider the Zilog Z8001. The Z8001 is a device that uses multiplexing to preserve small packaging. All address data lines are multiplexed as with the 8086. Thus, offboard latches and transceivers are required. There is then some peripheral delay associated with any system configuration. The Z8001 is basically the same signaling as the 8086; it uses an address strobe signal to latch valid addresses, and it uses a data strobe signal to request information back from the memory elements. We will consider a similar system as was described for the 8086-2 and apply that to the Z8001. As before, the first step is to calculate the basic memory cycle time for the Z8001. This is calculated in the equation given in Figure 1. The basic cycle time is 400 ns. We can then apply the same system configuration; that is, two levels of buffering, with an 8205 decoder. Going through a similar exercise indicates to us that the Z8001 requires 400 minus 88 minus 18 ns for the 8205 decoder. This gives it a t_{CE} requirement of 294 ns. This is a high performance requirement for an EPROM device. It indicates also that the Z8001 places a stringent requirement on the memory, and leaves less margin for the system designer to use. It also indicates that more stringent control is required of the memory device to insure no bus contention, and to insure that data arrives within the prescribed amount of time at the microprocessor.

In consideration of the MC68000, the use of the device is somewhat different with regard to addressing peripheral circuitry. The 68000 uses a technique which requires that a peripheral device signals the CPU when it is placing valid data onto the data bus. The 68000 does not use multiplexing, thus the package is quite large, but we have the advantage of not having to demultiplex the bus as it comes off the microprocessor. Should the microprocessor be designed correctly, this could give us quite an advantage in terms of memory cycle requirements. However, looking at the specifications of the 68000, it appears that the cycle requirements are even more stringent than on the 8086-2. These requirements are on the order of 275 ns. Thus we see that the demultiplexing of the address/data bus has not gained much in terms of removing some of the burden from the memory devices. We can go through a similar set of calculations as we did for the -2 and the Z8001. Doing so indicates that the MC68000 requires on the order of 170 ns for the memory device. The MC68000 uses what is called a data transfer acknowledge signal to indicate to the microprocessor that valid data is placed on the bus. This signal is typically taken from the chip enable signal that is provided to the EPROM but the timing of this places a very stringent requirement on t_{CE} . Since the 68000 does not lend itself nicely to two line control as a

HOTA
RESB

...and

ement



rix

N_____

en m

day.

RE-2 2732A RELIABILITY ENGINEERING EVALUATION REPORT

SUMMARY

Qualification testing of INTEL's high performance HMOS-E 32K EPROM has established HMOS as a reliable process for the fabrication of UV erasable EPROMs. High temperature dynamic lifetesting was used to evaluate the long term failure rate. At 55°C and a 60% UCL (user confidence level), a failure rate of .032%/1000 hours was observed based upon 1.3 million device hours of 125°C lifetesting.

RELIABILITY TESTING AND RESULTS

Five categories of testing were used to assure the electrical reliability of the 2732A:

1. High Temperature Dynamic Lifetest
2. High Temperature Reverse Bias (HTRB)
3. High Temperature Storage
4. Temperature Cycling
5. Low Temperature Lifetest

High Temperature Dynamic Lifetest—This test is used to accelerate failure mechanisms by operating the devices at an elevated temperature of 125°C. During the test the memory is sequentially addressed and the outputs are exercised, but not monitored or loaded. A checkerboard data pattern is used to simulate random patterns expected during actual use. Results of lifetesting on the 2732A are

shown in Table 1 along with the failure analysis. In order to best determine long term failure rate all devices used for lifetesting are subjected to standard INTEL screening. The 48 hour burn-in results measure infant mortality and are not included in the failure rate calculation.

Failure rate calculations are shown in Table 2 for each relevant activation energy. Failure rate calculations are made using the appropriate energy^{1,2,3} and the Arrhenius Plot as shown in Figure 1*. The total equivalent device hours at a given temperature can be determined. The failure rate is then calculated by dividing the numbers of failures by the equivalent device hours and is expressed as a %/1000 hours. The failure rate is adjusted by a factor related to the number of device hours using a chi-square distribution to arrive at a confidence level associated failure rate. A conservation estimate of the failure rate is obtained by including the zero based failure rate for 0.3eV failures. A failure rate of 0.032%/1000 hours at 55°C and 0.064%/1000 hours at 70°C using 60% UCL are determined for the 2732A. Devices for the other stresses received a 168 hour lifetest prior to stressing.

*The activation energies for various failure mechanisms are listed in Table 3.

Table 1. Lifetest Results

Burn-in 48HR	125°C Dynamic				150°C HTRB		
	168HR	500HR	1K HR	2K HR	168HR	500HR	1K HR
0/293	0/293	0/112	0/112	0/86	0/50	0/50	0/50
0/306	0/306	2/138 E	0/136	—	0/47	0/47	—
0/94	0/94	0/32	0/32	0/32	—	—	—
1/80 A	0/79	0/42	0/42	0/42	—	—	—
1/102 B	0/101	0/41	0/41	1/41 K	—	—	—
0/355	0/355	0/100	0/100	—	0/98	0/98	—
1/486 C	1/485 D	0/88	0/88	—	—	—	—
0/191	0/191	0/191	1/191 F	—	—	—	—
1/426 J	0/21	0/21	0/21	—	1/21 G	0/20	0/20
0/336	0/336	0/252	0/252	—	1/83 H	1/82 I	0/81
Totals	4/2669	1/2261	2/1017	1/1015	2/299	1/297	0/151

A = Part array unprogrammed
 B = Single bit charge retention
 C = Single bit charge retention
 D = Single bit charge retention (0.6eV)
 E = 2 each single bit charge retention (0.6eV)
 F = Single bit charge retention (0.6eV)

G = Multi edge bit charge retention, contamination
 H = Input leakage, contamination
 I = Input leakage, contamination
 J = Single bit charge retention
 K = Single bit charge retention (0.6eV)

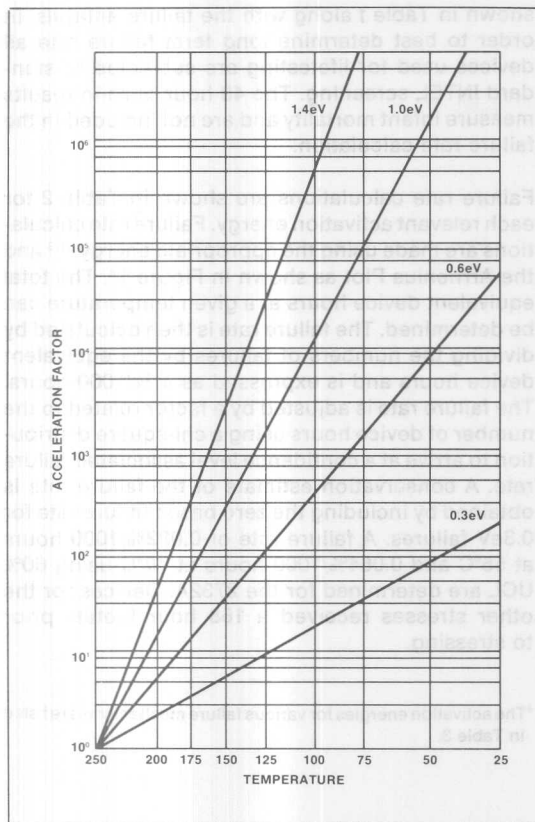


Figure 1. Arrhenius Plot

High Temperature Reverse Bias (HTRB)—This test is performed at 150°C and is effective in testing for leakage failures, device parameter drift, and data retention. HTRB results are included in the lifetest summary but are not used for the failure rate calculation. Three failures were observed in 299 units tested due to contamination.

High Temperature Storage—Another common test is high temperature storage in which devices are subjected to 250°C with no applied bias. This test is used to detect mechanical reliability problems (e.g., bond integrity), process stability, and data retention. Results from this test are shown in Table 4. Thirteen failures were observed from 532 devices tested due to single bit charge loss or contamination.

Temperature Cycle—This test consists of cycling the temperature of the chamber housing the devices from -65°C to +150°C. This test is used to detect mechanical reliability problems and microcracks. Results are shown in Table 4. No rejects were found on 81 devices.

Low Temperature Lifetest—This test is performed to detect the effects of hot electron injection into the gate oxide² as well as package related failures (corrosion of internal metal lines, etc.). This test is performed at -10°C with $V_{CC} = 5.0$ volts. Results are shown in Table 4. One reject in 100 devices was found for single bit charge retention.

Table 2. Failure Rate Predictions

Actual Device Hours @ 125°C	Equivalent Device Hours			Failure Rate/1000 Hours (60%UCL)		
	Ea	55°C	70°C	#Fail	55°C	70°C
1.32 x 10 ⁶	0.6eV	3.6 x 10 ⁷	1.5 x 10 ⁷	5	0.017%	0.041%
1.32 x 10 ⁶	0.3eV	6.9 x 10 ⁶	4.5 x 10 ⁶	0	0.015%	0.023%
Combined Failure Rate Fits*					0.032% 320.	0.064% 640.

*FIT — Failures in time, 1 FIT=1 Failure per 10⁹ Device Hrs.

Table 3.
Failure Mechanism Activation Energies
Relevant to EPROMs

Failure Mode	Activation Energy
Random bit charge gain/loss	.6eV
Oxide breakdown	0.3eV
Silicon defects	0.3eV
Contamination	1.0–1.4eV

dress. Each output is 128 rows by 32 columns and the outputs are arranged as shown.

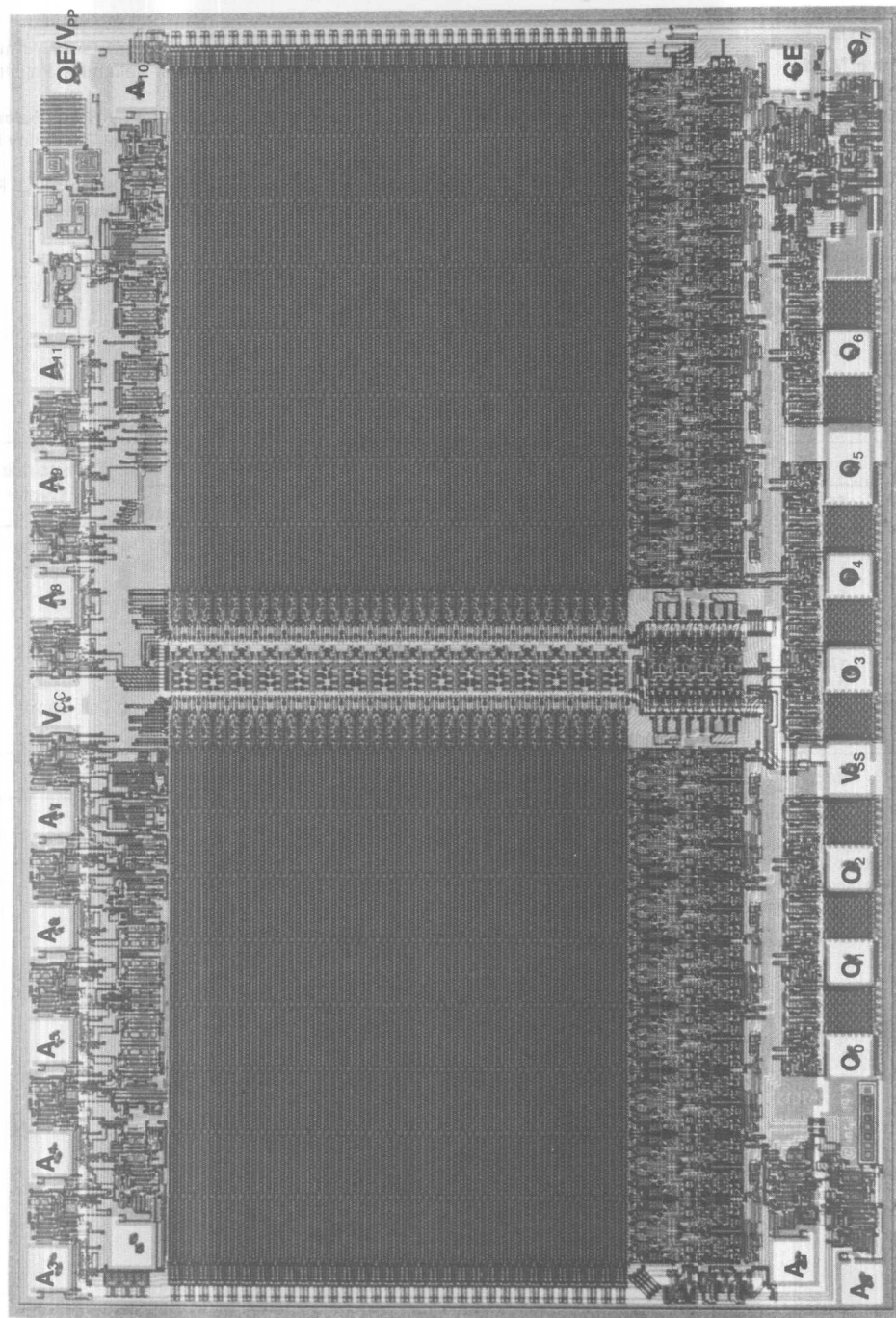
1. S. Rosenberg, D. Crook, B. Euzent, "16th Annual Proceedings of the International Reliability Physics Symposium," pp 19–25, 1978.
2. J. Caywood, B. Euzent, B. Shiner, "Data Retention in EPROMs," 1980 IEEE International Reliability Physics Symposium.
3. S. Rosenberg, B. Euzent, "HMOS Reliability" Reliability Report RR-18, INTEL Corporation, 1979.

Bit Map—Figure 2 shows a bit map for the 2732A. The bit map shows the actual physical location of each bit corresponding to its row and column ad-

Table 4. Stress Results

	250°C Bake			–10°C Dynamic		Temperature Cycle
	48 HR	168 HR	500 HR	500 HR	1K HR	
0/50	0/50	0/50	0/50	0/25	0/25	0/20
1/48 A	0/47	0/47	1/47 E	1/25 H	0/24	0/20
0/50	0/50	0/50	1/50 F	—	—	—
0/25	0/25	0/25	0/25	—	—	—
1/49 B	0/48	0/48	2/48 G	—	—	—
1/49 C	0/48	0/48	0/48	0/25	0/25	0/20
0/99	—	—	—	0/25	—	—
3/75 D	0/72	0/72	0/72	—	—	0/21
2/87 I	0/85	0/85	1/85 J	—	—	—
Totals	8/532	0/425	5/425	1/100	0/74	0/81

A, B, C, D, H = Single bit charge retention
 E, F, G = Multi edge bit charge retention, contamination
 I = 1 ea. single bit charge retention
 1 ea. input leakage, contamination
 J = Input leakage, contamination



TRUTH TABLE

A ₁	A ₀	B ₁	B ₀	C ₀	S ₀	S ₁	S ₂	S ₃	C ₄
0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	0	1	0	0	1	0	0	0
0	0	0	1	1	1	1	0	0	0
0	0	1	0	0	1	0	1	0	0
0	0	1	0	1	0	0	1	1	0
0	0	1	1	0	0	1	1	0	0
0	0	1	1	1	1	0	0	1	0
0	1	0	0	0	1	1	0	0	0
0	1	0	0	1	0	0	1	0	0
0	1	0	1	0	0	1	1	0	0
0	1	0	1	1	1	0	0	1	0
0	1	1	0	0	1	0	1	1	0
0	1	1	0	1	0	1	0	0	1
0	1	1	1	0	0	1	1	0	1
0	1	1	1	1	1	0	0	1	1
1	0	0	0	0	1	1	1	0	0
1	0	0	0	1	0	0	1	1	0
1	0	0	1	0	0	1	0	1	1
1	0	0	1	1	1	0	1	0	1
1	0	1	0	0	1	0	1	1	1
1	0	1	0	1	0	1	0	0	1
1	0	1	1	0	0	1	1	0	1
1	0	1	1	1	1	0	0	1	1
1	1	0	0	0	1	1	1	1	0
1	1	0	0	1	0	0	1	1	1
1	1	0	1	0	0	1	0	0	1
1	1	0	1	1	1	0	1	1	1
1	1	1	0	0	1	0	1	1	1
1	1	1	0	1	0	1	0	0	1
1	1	1	1	0	0	1	1	0	1
1	1	1	1	1	1	0	0	1	1

LOGIC DIAGRAM

The logic diagram shows four full-adder blocks. The first block takes inputs A₀ and B₀ and produces sum output S₀ and carry output C₁. The second block takes inputs A₁ and B₁ and produces sum output S₁ and carry output C₂. The third block takes inputs A₂ and B₂ and produces sum output S₂ and carry output C₃. The fourth block takes inputs A₃ and B₃ and produces sum output S₃ and carry output C₄. The carry inputs are C₀ for the first block, and C₁, C₂, C₃ for the subsequent blocks.

TIMING DIAGRAM

The timing diagram shows waveforms for inputs A₁, A₀, B₁, B₀, carry input C₀, and outputs S₀, S₁, S₂, S₃, and carry output C₄ over 16 clock cycles. The inputs are binary signals. The outputs are binary signals that change at the rising edge of the clock.

Figure 2B. 2732A Bit Map, $A_{11} = 1$ Shown

Type	No. of Bits	Organization	No. of Pins	Output	Access (ns)	Dissipation (mW)	Range (°C)	Supply (V)
3628A-1	8192	1024x8	24	T.S.	50	998	0 to 75	5V ± 10%
3628A-3	8192	1024x8	24	T.S.	70	998	0 to 75	5V ± 10%
3628A-4	8192	1024x8	24	T.S.	90	998	0 to 75	5V ± 10%
3636	16384	2048x8	24	T.S.	80	998	0 to 75	5V ± 10%
3636-1	16384	2048x8	24	T.S.	65	998	0 to 75	5V ± 10%
3636B-1	16384	2048x8	24	T.S.	35	998	0 to 75	5V ± 10%
3636B-2	16384	2048x8	24	T.S.	45	998	0 to 75	5V ± 10%
M3636	16384	2048x8	24	T.S.	80	998	-50 to 125	5V ± 5%

BIPOLAR PROM CROSS REFERENCE

Part Number	Prefix and Manufacturer	Organization	Intel Part Number	
			Direct Replacement	For New Designs
82S181	N-Signetics	1024x8	3628A-3	3628A-1
82S191	N-Signetics	2048x8	3636	3636B-1
82S191	S-Signetics	2048x8	M3636	

MOS EPROM FAMILY

Type	No. of Bits	Organization	No. of Pins	Output	Maximum Access (ns)	Maximum Power Dissipation (mW)	Operating Temperature Range (°C)	Power Supply (V)
2716	16384	2048x8	24	T.S.	450	525/132	0 to 70	5V ± 5%
2716-1	16384	2048x8	24	T.S.	350	550/138	0 to 70	5V ± 10%
2716-2	16384	2048x8	24	T.S.	390	525/132	0 to 70	5V ± 5%
2716-5	16384	2048x8	24	T.S.	490	525/132	0 to 70	5V ± 5%
2716-6	16384	2048x8	24	T.S.	650	525/132	0 to 70	5V ± 5%
12716	16384	2048x8	24	T.S.	450	605/165	– 40 to 85	5V ± 5%
M2716M	16384	2048x8	24	T.S.	450	635/165	– 55 to 125	5V ± 10%
M2716	16384	2048x8	24	T.S.	450	635/165	– 55 to 100	5V ± 10%
2732	32768	4096x8	24	T.S.	450	790/185	0 to 70	5V ± 5%
2732-4	32768	4096x8	24	T.S.	390	790/185	0 to 70	5V ± 5%
2732-6	32768	4096x8	24	T.S.	550	790/185	0 to 70	5V ± 5%
2732A	32768	4096x8	24	T.S.	250	790/185	0 to 70	5V ± 5%
2732A-2	32768	4096x8	24	T.S.	200	790/185	0 to 70	5V ± 5%
2732A-3	32768	4096x8	24	T.S.	300	790/185	0 to 70	5V ± 5%
M2732	32768	4096x8	24	T.S.	450	825/250	– 55 to 100	5V ± 10%
M2732 S8416	32768	4096x8	24	T.S.	550	825/250	– 55 to 125	5V ± 10%
2764	65536	8192x8	28	T.S.	250	790/185	0 to 70	5V ± 5%
2764-2	65536	8192x8	28	T.S.	200	790/185	0 to 70	5V ± 5%
2764-3	65536	8192x8	28	T.S.	300	790/185	0 to 70	5V ± 5%
2764-4	65536	8192x8	28	T.S.	450	790/185	0 to 70	5V ± 5%

MOS E²PROM FAMILY

2816	16384	2048x8	24	T.S.	250	495/135	0 to 70	5V ± 5%
2816-3	16384	2048x8	24	T.S.	350	495/135	0 to 70	5V ± 5%
M2816	16384	2048x8	24	T.S.	300	825/195	– 55 to 125	5V ± 10%

Memory System Design Information

4

[illegible]

INTRODUCTION

Complex electronic systems require the utmost in reliability. Especially when the storage and retrieval of critical data demands faultless operation, the system designer must strive for the highest reliability possible. Extra effort must be expended to achieve this high reliability. Fortunately, not all systems must operate with these ultra reliability requirements.

The majority of systems operate in an area where system failure ranges from irritating, such as a video game failure, to a financial loss, such as a misprinted check. While these failures are not hazardous, reliability is important enough to be designed into the system.

A memory system is one of the system components for which reliability is important. Also, it is one of the few system components which can be altered to greatly enhance its reliability. The purpose of this report is to examine different methods of error encoding, especially Error Correction Codes (ECC), to increase the reliability of the memory system.

SYSTEM RELIABILITY

Individual device reliability is the foundation of memory system reliability. Reliability is expressed as mean time between failures (MTBF) of a system is a function of the number of devices and the device failure rate. Failure rate of the memory device can be obtained from the reliability report on the specific device. MTBF of the device is:

$$T_D = \frac{1}{\lambda} \quad [1]$$

where T_D = MTBF of the device

λ = device failure rate (%/1000 hrs)

and MTBF of the system is approximately:

$$T_S = \frac{T_D}{D} \quad [2]$$

where T_S = MTBF of the system

D = number of devices in the system

As the number of devices required to construct a system becomes larger, the system MTBF becomes smaller.

A plot of system MTBF as a function of the number of memory devices is shown in Figure 1 for different failure rates. Included for reference are the failure rates of the Intel® 2104A 4Kx1 RAM and the Intel® 2117 16Kx1 RAM. Using RAMs which are organized one bit wide, the amount of devices required for a system is calculated by multiplying the number of words by the word length

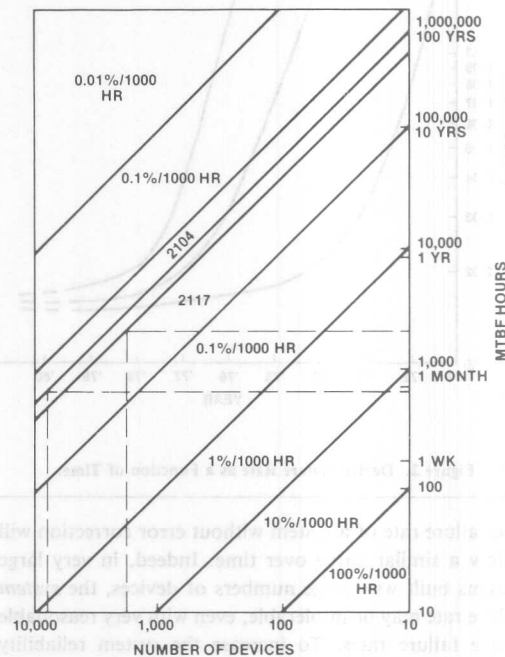


Figure 1. System Reliability vs Number of Devices

and dividing by the size of the RAM. To illustrate, assume a 1 megaword memory system with a word width of 32 bits, implemented with Intel® 2104A 4Kx1 RAMs. The number of required devices is:

$$D = \frac{1,048,576 \times 32}{4,096} = 8,192 \text{ devices}$$

Prediction of failure for this system, shown in Figure 1, is 667 hours or 28 days — assuming continuous use and worst case temperature.

Equation 2 showed that system MTBF is increased when fewer devices are used. A one megaword memory having 32 bit wide words can be constructed with Intel 2117 16K RAMs. In this case one fourth as many devices are required — 2048 devices. From Equation 2, the expected MTBF should be four times as large — 2668 hours. It is not. The failure rate from Figure 1 for this system is 2000 hours. Different device failure rates account for this difference. The failure rate of the 16K is not yet equal to that of the 4K. Memory device reliability is a function of time as shown in Figure 2. Reliability improvement often is a result of increased experience in manufacturing and testing. In time, the failure rate of the 16K will reach that of the 4K and one fourth as many devices will result in a system MTBF approximately four times better.

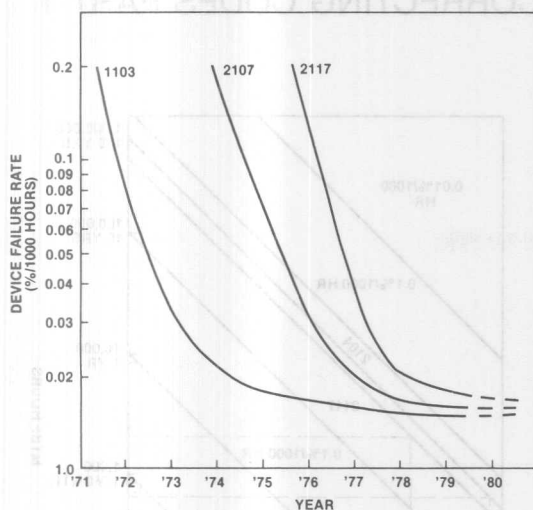


Figure 2. Device Failure Rate as a Function of Time.

The failure rate of a system without error correction will follow a similar curve over time. Indeed, in very large systems built with large numbers of devices, the *system* failure rate may be intolerable, even with very reasonable *device* failure rates. To increase the system reliability beyond the device reliability, *redundancy coding techniques* have been developed for detecting and correcting errors.

REDUNDANCY CODES

Redundancy codes add bits to the data word to provide a validity check on the entire word. These additional bits, used to detect whether or not an error has occurred, are called encoding bits. With M data bits and K encoding bits, the encoded word width is N bits. Shown in Figure 3 is the form of the encoded word.

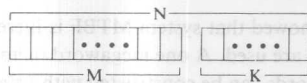


Figure 3. Encoded Word Form

Mathematically, N is related to M and K by:

$$N = M + K \quad [3]$$

where N = number of bits in the encoded word

M = number of data bits

K = number of encoding bits

Exactly how K is related to M , and the number of required K bits depends on several factors which will be described later.

One measure of a code is its efficiency. Efficiency is the ratio of the number of bits in the encoded word to the number of bits of data:

$$E = \frac{N}{M}$$

Substituting $N = M + K$:

$$E = \frac{M + K}{M} \quad [4]$$

where E = efficiency

All of the data are contained in the M bits. The K bits contain no data, only validity checks. To maximize the amount of data in the encoded word, the number of K bits must be minimized. Examination of Equation 4 shows that the minimum value of K is zero. With K equal to zero, the efficiency is unity. Efficiency is maximized, but the word has no encoding bits. Therefore, it has no capability to detect an error.

As an example, consider a two bit word. It can assume 2^2 or 4 states, which are:

State 1	00
State 2	01
State 3	10
State 4	11

Figure 4. All States of a Two-Bit Word

All possible states have been used as data; consequently any error will cause the error state to be identical to a valid data state.

The mechanics of the encoding bits create encoded words such that every valid encoded word has a set of error words which differ from all valid encoded words. When an error occurs, an error word is formed and this word is recognized as containing invalid data.

By adding one K bit to the two bit word error detection becomes possible. The value of the K bit will be such that the encoded word has an odd number of ONES. As will be explained later, this technique is "odd" parity.

The sum of the ONES in a word is the *weight* of the word. Parity operates by differentiating between odd and even weights. The encoded word will always have an odd weight as a result of having an odd number of ONES.

If a single bit error occurs, one bit in the encoded word will change state and the word will have an even weight. Then in this example, all encoded states with an even weight — an even number of ones — are error states.

The value of the encoding bit or parity bit is found by counting the number of ones — calculating the weight — and setting the value of K to make the weight of the encoded word odd. Referring to Figure 4, State 1 was 00,

the weight of this word is 0, so K is set to 1 and the weight of the encoded word is odd. State 2 is 01, the weight is odd already, so K is set to 0. The weight of State 3 is identical to that of State 2 so K is again set to 0. Finally, State 4 has an even weight ($1 + 1 = 2$), thus K is 1. The encoded states of the two bit data word are listed in Figure 5.

	Data	Encoding Bit
State 1	00	1
State 2	01	0
State 3	10	0
State 4	11	1
	M	K

Figure 5. Code Bits for All Possible States of a Two-Bit Word

To illustrate the error detection, Figure 6a lists all states of the encoded data word and all possible single bit errors. Because the encoded word is 3 bits long, there are only 3 possible single bit errors for each encoded state.

	A	B	C	D
Encoded States	001	010	100	111
Error States	000	000	000	011
	011	011	101	101
	101	110	110	110

Figure 6a. All Possible Single-Bit Errors

Notice that every error state has an even weight, while the valid encoded states have odd weights.

Converting all the values of these states to decimal equivalents makes the errors more obvious as shown in Figure 6b.

Valid States	1	2	4	7
	0	0	0	
Error States	3	3		3
	5		5	5
		6	6	6

Figure 6b. Decimal Representation of Errors

No error state is the same as any valid encoded state. Identical error states can be found in several columns. The fact that some error states are identical prevents identification of the bit in error, and hence correction is impossible. Importantly though, error detection has occurred.

Figure 6a demonstrates another property of codes. Every error state differs from its valid encoded state by one bit, whereas each of the encoded states differs from the others by two bits. Examine the encoded states labeled B and D in Figure 6a and shown in Figure 7.

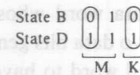


Figure 7. Bit Difference.

These two states have two bit positions which differ. This *difference* is defined as *distance* and these two states have a distance of two. Distance, then, is the number of bits that differ between two words. The encoded words have a minimum distance of two. Longer encoded words may have distances greater than two but never less than two if error detection is desired. The error states have a minimum distance of one from their valid encoded state.

A minimum distance of two between encoded states is required for error detection. A re-examination of a word with no encoding bits shows that the states have a minimum distance of 1 (see Figure 8). No error detection is possible because any single bit error will result in a valid word.

State 1	00
State 2	01
State 3	10
State 4	11

Figure 8. Minimum Distance of a Two-Bit Word

PARITY

A minimum distance of two code is implemented with Parity. Refer to previous section for an explanation. Parity is generated by exclusive-ORing all the data bits in the word, which results in a parity bit. This parity bit is the K encoding bit of the word. If the word contains M data bits, the parity bit is:

$$C = b_1 \oplus b_2 \oplus b_3 \oplus \dots \oplus b_m$$

where C = parity bit

b = value in the bit position

The parity bit combines with the original data bits to form the encoded word as shown in Figure 9. Encoded words always have either "odd" parity, which is an odd number of 1s (an odd weight) or "even" parity which is an even number of 1s (an even weight). Odd and even parity are never intermixed, so that the encoded words all have either odd or even parity — never both.

When the encoded word is fetched, the parity bit is removed from the word and saved. A new parity bit is generated from the M bits. Comparing this new parity bit with the stored parity bit determines if a single bit error has occurred.

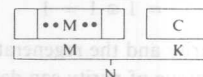


Figure 9. Encoded Word Form

Consider the two bit data word whose value is "01." Exclusive-NORing the two data bits generates a parity bit which causes the encoded word to have odd parity:

$$\bar{C} = 0 \oplus 1$$

$$\bar{C} = 0$$

The encoded word becomes:

$$\begin{array}{cc} M & K \\ \hline 0 & 1 \\ \hline \end{array} \quad \begin{array}{l} \text{parity} \\ \text{LSB of data} \end{array}$$

Assume that an error occurs and the value of the word becomes "110." Stripping off the parity bit and generating a new parity bit:

$$\text{transmitted parity} = 0$$

$$\text{transmitted word} = 11$$

$$\text{new parity of transmitted word} = 1 \oplus 1 = 1$$

$$\text{generated parity} \neq \text{transmitted parity}$$

Note that the error could have occurred in the parity bit and the final result would have been the same. An error in the encoding bit as well as in the data bits can be detected.

Although parity detects the error, no correction is possible. This is because each valid word can generate the same error state. Illustration of this is shown in Figure 10.

Correct Word with Parity	Possible Single Bit Error
0 0 1	0 1 1
1 1 1	0 1 1
0 1 0	0 1 1

Figure 10. Possible Errors

Each of the errors is identical to the others and reconstruction of the original word is impossible.

Parity fails to detect an *even* number of errors occurring in the word. If a double bit error occurs, no error is detected because two bits have changed state, causing the weight of the word to remain the same.

Using the encoded word "010" one possible double bit error (DBE) is:

$$\begin{array}{cc} 1 & 1 & 1 \\ \hline & & \text{parity} \end{array}$$

Checking parity:

$$\bar{C} = 1 \oplus 1 = 1$$

The transmitted parity and the regenerated parity agree. Therefore the technique of parity can detect only an *odd* number of errors.

In summary, single bit parity will detect the majority of errors, but cannot be used to correct errors. Using parity introduces a measure of confidence in the system. Should a single bit error occur, it will be detected.

ERROR CORRECTION

Classical texts on error coding contain proofs showing that a minimum distance of three between encoded words is necessary to correct errors. While this fact does not describe the code, it does give an indication of the form of the code.

Correcting errors is not as difficult as it first appears. As a result of a paper published by R. W. Hamming on error correction the most widely used type of code is the "Hamming" code. Using the same technique as parity, Hamming code generates K encoding bits and appends them to the M data bits. As shown in Figure 11, this N bit word is stored in memory.

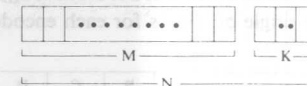


Figure 11. Encoded Word Form

Thus far the mechanism is similar to parity. The only difference is the number of K bits and how they relate to the M data bits.

When the word is read from memory, a new set of code bits (K') is generated from the M' data bits and compared to the fetched K encoding bits. Comparison is done by exclusive-ORing as shown in Figure 12. Like parity the result of the comparison — called the syndrome word — contains information to determine if an error has occurred. Unlike parity, the syndrome word also contains information to indicate which bit is in error.

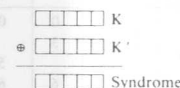


Figure 12. Syndrome Generation

The *syndrome word* is therefore K bits wide. The syndrome word has a range of 2^K values between 0 and $2^K - 1$. One of these values, usually zero, is used to indicate that no error was detected, leaving $2^K - 1$ values to indicate which of the N bits was in error. Each of these $2^K - 1$ values can be used to uniquely describe a bit in error. The range of K must be equal to or greater than N. Mathematically, the formula is:

$$2^K - 1 \geq N$$

$$\text{but } N = M + K$$

$$\text{and } 2^K - 1 \geq M + K \quad [5]$$

Equation 5 gives the number of K bits needed to correct a single bit error in a word containing M data bits. Ranges of M for various values of K are calculated and listed in Table I.

K	Single Correct/ Single Detect		Single Correct/ Double Detect	
	$\leq M \leq$		$\leq M \leq$	
4	4	11	1	3
5	12	26	4	10
6	27	57	11	25
7	58	120	26	56
8	121	245	57	119

Table I.

Range of M for Single Correct/Single Detect or Double Detect Codes for Values of K

To detect and correct a single bit error in a 16 bit data word, five encoding bits must be used. As a result, the total number of bits in the encoded word is 21 bits.

Efficiencies of single detect — parity — and single detect/single correct codes as a function of the number of data bits are shown in Figure 13. For large values of M, the efficiency of single detect/correct is approximately equal to that of the single detect code — parity.

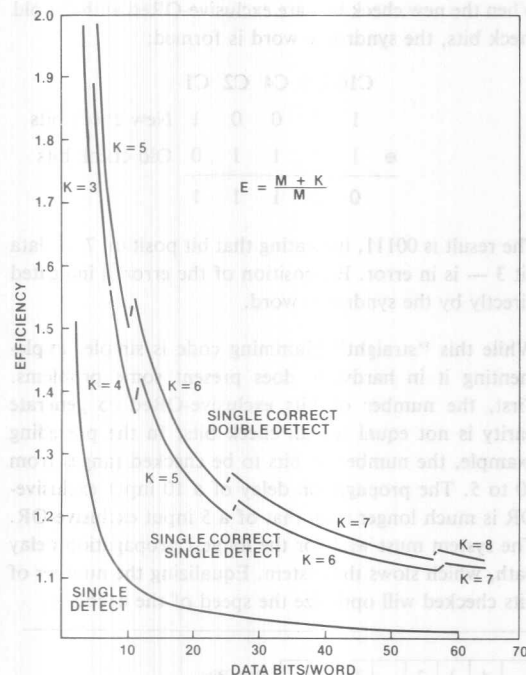


Figure 13. Code Efficiency vs Data Word Size

CODE DEVELOPMENT

Contained in the syndrome word is sufficient information to specify which bit is in error. After decoding this information, error correction is accomplished by inverting the bit in error. All bits, including the encoding bits — called check bits — are identified by their positions in the word.

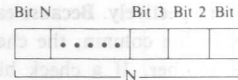


Figure 14. Positional Representation of Bits in the Word

Bits in the N bit word are organized as shown in Figure 14. Bit numbers shown in decimal form are converted to binary numbers. From equation 5, this binary number will be K bits wide. In Figure 15 is an example using a 16 bit data word. Because there are 16 data bits, M equals 16, K equals 5 and N equals 21. Shown in Figure 15 the word is binary equivalent of the position. Notice that where the M and the K bits are located is not yet specified.

Bit 21	Bit 20	Bit 19	Bit 18	Bit 17	Bit 16	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit Position Value
N																					
1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	2 ⁰ LSB
0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	2 ¹
1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	2 ²
0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	2 ³
1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2 ⁴ MSB

Figure 15. Binary Value of Bit Position.

The syndrome word is the difference between the fetched check bits and the regenerated check bits. Identification of the bit in error by the syndrome word is provided by the binary value of the bit position. The syndrome word is generated by exclusive-ORing the fetched check bits with the regenerated check bits. Any new check bits that differ from the old check bits will set 1s in the syndrome word. To identify bit 3 as a bit in error, the syndrome word will be 00011, which is the binary value of the bit position. Weight is determined only by the 1s in the bit position chart in Figure 15, so they are replaced with an X and the 0s are deleted. The result is shown in Figure 16.

Bit 21	Bit 20	Bit 19	Bit 18	Bit 17	Bit 16	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1
N																				
X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
XX				XX				XX				XX				XX				C1
XX		XXXX						XXXX						XX						C2
XXXXXXXXXX																		C4		
																		C8		
																		C16		

Figure 16. Relationship of Data Bits and Check Bits.

row shown in Figure 16. Five bit positions — 1, 2, 4, 8, and 16 — have only one X in their columns. The corresponding check bits are in these respective locations. Check bit C1 is stored in Bit Position 1, C2 is stored in Bit Position 2, and C4, C8, and C16 are stored in positions 4, 8, and 16 respectively. Because each of these positions has one X in the column, the check bits are independent of one another. If a check bit fails, the syndrome word will contain a single “1.” A data bit failure will be identified by two or more “1s” in the syndrome word.

The data bits are filled in the positions between the check bits. The least significant bit (LSB) of data is located in position 3.

Data Bit 2 is stored in position 5 — position 4 is a check bit. Figure 17 shows the positions of data bits and check bits for sixteen bits of data.

When the check bits are generated for storage, bits 1, 2, 4, 8, and 16 are omitted from the generation circuitry because they do not yet exist, being the result of generation.

Parity check on the specified bits is used to generate the check bits. Each check bit is the result of exclusive-ORing the data bits marked with an “X” in Figure 18. Check bits are generated by these logic equations:

$$C1 = M1 \oplus M2 \oplus M4 \oplus M5 \oplus M7 \oplus M9 \oplus M11 \oplus M12 \oplus M14 \oplus M16$$

$$C2 = M1 \oplus M3 \oplus M4 \oplus M6 \oplus M7 \oplus M10 \oplus M11 \oplus M13 \oplus M14$$

$$C4 = M2 \oplus M3 \oplus M4 \oplus M8 \oplus M9 \oplus M10 \oplus M11 \oplus M16 \oplus M16$$

$$C8 = M5 \oplus M6 \oplus M7 \oplus M8 \oplus M9 \oplus M10 \oplus M11$$

$$C16 = M12 \oplus M13 \oplus M14 \oplus M15 \oplus M16$$

How the Hamming code corrects an error is best shown with an example. In this example, a data word will be assumed, check bits will be generated, an error will be forced, new check bits will be generated, and the syndrome word will be formed. Assuming the 16-bit data word

0101 0000 0011 1001

Check bits are generated by overlaying the data word on the Hamming Chart of Figure 16 and performing an odd parity calculation on the bits matching the “Xs.”

16	15	14	13	12		11	10	9	8	7	6	5		4	3	2		1		
					C16								C8					C4	C2	C1
21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1

Data Bits

Check Bits

Position

Figure 17. Data and Check Bit Positions in the Encoded Word.

by “1s,” only columns containing “1s” are circled for identification. The check bits are the result of odd parity generated on the rows. For example, the C1 row has three “Xs” circled; therefore C1 is 0 to keep the row parity odd. In this example, all other rows contain an even number of circled “Xs;” therefore the remaining check bits are “1s.” These check bits are incorporated into the data word, forming the encoded word. Performing this function, the 21 bit encoded word is:

C16 C8 C4 C2 C1
0101 0 1 000 0011 1 100 1 1 1 0

Forcing an error with bit position 7 — data bit 4:

C16 C8 C4 C2 C1
0101 0 1 000 0011 1 000 1 1 1 0

A new set of check bits is generated on the error word as shown in Figure 18 and is:

C16 C8 C4 C2 C1
1 1 0 0 1

When the new check bits are exclusive-ORed with the old check bits, the syndrome word is formed:

C16	C8	C4	C2	C1	
1	1	0	0	1	New check bits
⊕	1	1	1	0	Old check bits
0	0	1	1	1	

The result is 00111, indicating that bit position 7 — data bit 3 — is in error. Bit position of the error is indicated directly by the syndrome word.

While this “straight” Hamming code is simple, implementing it in hardware does present some problems. First, the number of bits exclusive-ORed to generate parity is not equal for all check bits. In the preceding example, the number of bits to be checked ranges from 10 to 5. The propagation delay of a 10 input exclusive-OR is much longer than that of a 5 input exclusive-OR. The system must wait for the longest propagation delay path, which slows the system. Equalizing the number of bits checked will optimize the speed of the encoders.

Figure 18a. Hamming Chart.

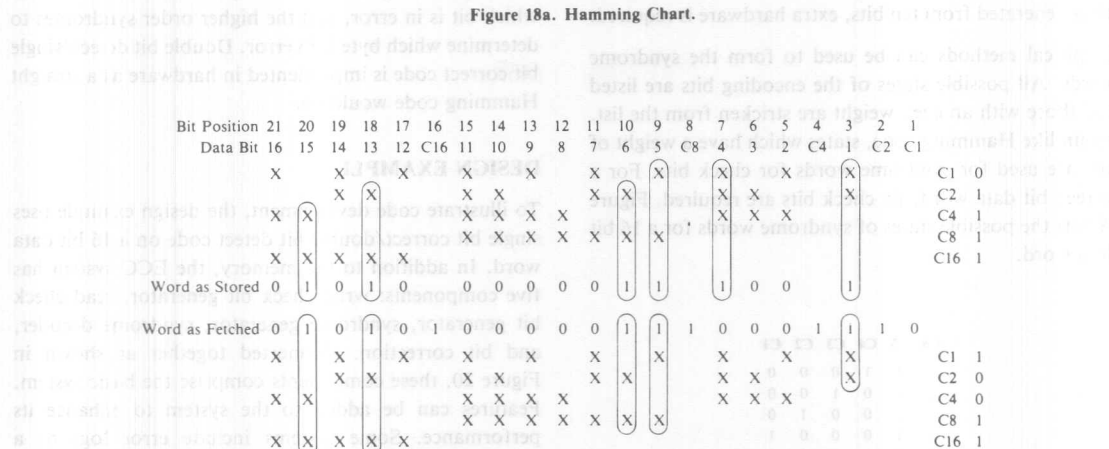


Figure 18b. Check Bit Generation.

Secondly, two bits in error can cause a correct bit to be indicated as being in error. For example, if check bits C1 and C2 failed, data bit 1 would be flagged as a bit in error.

Because of these two difficulties, the Error Correction Code (ECC) most commonly used is a "modified" Hamming code is most widely used which will detect double bit errors and correct single bit errors.

SINGLE BIT CORRECT/ DOUBLE BIT DETECT CODES

Modern algebra can be used to prove that a minimum distance of four is required between encoded words to detect two errors or correct a single bit error. An excellent text on this subject is *Error Correcting Codes* by Peterson and Weldon.

One possible double bit error is two check bits. Using straight Hamming code, the circuit would "correct" the wrong bit. Double error detection techniques — modified Hamming codes — prevent this by separating the encoded words by a minimum distance of four. As a result each data bit is protected by a minimum of three check bits, so that the syndrome word always has an odd weight. Therefore, even weight syndrome words cannot be used. When two check bits fail, the syndrome word has two "1s" or an even weight. Even weight is

detectable as a double bit error by performing a parity check on the syndrome word. If two data bits fail, again the syndrome word has an even weight — a detectable error.

Adding one additional check bit to the correction check bits provides the capability to detect double bit errors. The number of encoding or check bits required to detect double bit errors and correct single bit errors is:

$$2^M \leq \frac{2^{N-1}}{N}$$

Substituting M + K for N:

$$2^{K-1} \geq M + K \quad [6]$$

Equation 6 is similar to equation 5, which describes single bit correct and detect except for the left side of the inequality, which shows one additional encoding bit is required. For single bit detect and correct the left side of the inequality was 2^K . Table I also lists the ranges of M for values of K, for a direct comparison to single bit detect and single bit correct codes.

Figure 13 includes the efficiency curve for single bit correct/double bit detect (SBC/DBD) codes for values of M. As would be expected, because of the additional encoding bit the efficiency is slightly lower. For large values of M, the efficiency of this code approaches unity like the two other curves.

Syndrome words for the SBC/DBD code are developed like the straight Hamming code, except that syndrome words do not map directly to bit positions. The syndrome word has an odd weight and does not increment like straight Hamming code. In addition, implementation considerations can impose constraints. For example, the 74S280 parity generator is a nine input device. If a check bit is generated from ten bits, extra hardware is required.

Empirical methods can be used to form the syndrome words. All possible states of the encoding bits are listed and those with an even weight are stricken from the list. Again like Hamming code, states which have a weight of one are used for syndrome words for check bits. For a sixteen bit data word, six check bits are required. Figure 19 lists the possible states of syndrome words for a 16 bit data word.

C6	C5	C4	C3	C2	C1
1	1	1	0	0	0
1	1	0	1	0	0
1	1	0	0	1	0
1	1	0	0	0	1
1	0	1	1	0	0
1	0	1	0	1	0
1	0	1	0	0	1
1	0	0	1	1	0
1	0	0	1	0	1
1	0	0	0	1	1
0	1	1	1	0	0
0	1	1	0	1	0
0	1	1	0	0	1
0	1	0	1	1	0
0	1	0	1	0	1
0	1	0	0	1	1
0	0	1	1	1	0
0	0	1	1	0	1
0	0	1	0	1	1
0	0	0	1	1	1
0	0	0	0	0	1
0	0	0	0	1	0
0	0	0	1	0	0
0	0	1	0	0	0
0	1	0	0	0	0
1	0	0	0	0	0

Figure 19. Possible Syndrome Words

In Figure 19 only twenty syndrome words for data bits are listed, because the possible words with a weight of 5 were eliminated so that every data bit would have only three bits protecting it. This simplifies the hardware implementation. If there are more than 20 data bits, states with a weight of 5 must be used. All states listed in Figure 19 are valid syndrome words, so that the problem becomes one of selecting the optimum set of syndrome words. To minimize circuit propagation delay the number of data bits checked by each encoding bit should be as close as possible to all the others.

The syndrome words can be mapped to any bit position, providing that identical code generations are done at storage and retrieval times. Syndrome word mapping may be arranged to solve system design problems. For example, in byte oriented systems the lower order syndrome bits are identical, so that the circuit design may be simplified by using these syndromes to determine which bit is in error, and the higher order syndromes to determine which byte is in error. Double bit detect/single bit correct code is implemented in hardware as a straight Hamming code would be.

DESIGN EXAMPLE

To illustrate code development, the design example uses single bit correct/double bit detect code on a 16 bit data word. In addition to the memory, the ECC system has five components: write check bit generator, read check bit generator, syndrome generator, syndrome decoder, and bit correction. Connected together as shown in Figure 20, these components comprise the basic system. Features can be added to the system to enhance its performance. Some systems include error logs as a feature. Because the address of the error and the errors are known, the address and the syndrome word are saved in a non-volatile memory. At maintenance time this error log is read and the indicated defective devices are replaced. Being a basic design, this example does not include an error log.

Write check bits are generated when data are written into the memory, while read check bits are generated when data are read from the memory. Off-the-shelf TTL is used to implement the design. Check bits are generated by performing parity on a set of data bits, so that this function is performed by 74S280 9-bit parity generators. One parity generator for each check bit is required. Because the read and write check bit generations are the same, the circuits are similar. One minor difference should be noted. In this example, the check bit will be formed from parity on eight data bits. The 74S280 parity generator has nine inputs; therefore, the write check bit generator will have the extra input grounded while the read generator has as an input the fetched check bit. Developed directly in the read check bit generator is the syndrome bit, which saves one level of gating. Figure 21 shows the identical results of generating the syndrome bit by exclusive-ORing the fetched check bit with the regenerated check bit and forming the syndrome bit in the read check bit generator.

Implementing the syndrome generator word in this way reduces the circuit propagation delay by approximately 10 nanoseconds. This implementation imposes a restriction on the code to be used — the check bit must be formed from no more than eight data bits.

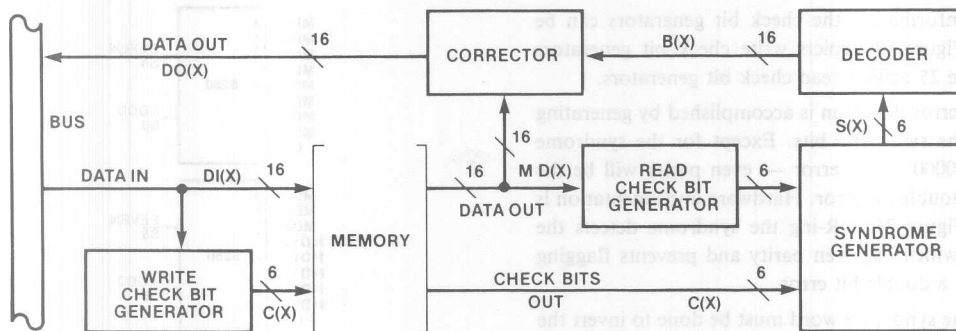


Figure 20. Block Diagram of ECC System.

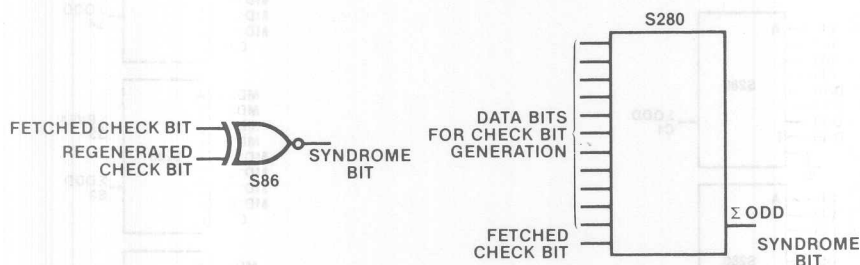


Figure 21. Syndrome Bit Generation.

Figure 19 listed the possible syndrome words for a 16 bit data word. These are relisted in Figure 22 with the syndrome words for the check bits and the zeros deleted.

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	C1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	C2
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	C3
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	C4
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	C5
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	C6

Figure 22. Possible Syndrome Words with Three Check Bits.

While there are twenty possibilities for syndrome words, only 16 are needed. Each row contains ten "1s" and each column contains three "1s." Four columns are eliminated but in a way that each row contains eight "1s." When the columns are matched to data bits, the "1s" in each row define inputs to the 74S280 parity generators for the given check bit. Eliminating the two columns from each end results in sixteen columns with each row having eight "1s." These remaining sixteen columns which match the data bits are rearranged in Figure 23 for convenience of printed circuit board layout and assigned to the data bits. The syndrome words for check bits are also shown for complete code development.

Data Bit																						
M16	M15	M14	M13	M12	M11	M10	M9	M8	M7	M6	M5	M4	M3	M2	M1	C1	C2	C3	C4	C5	C6	
X				X	X	X		X		X			X		X	X					C1	
	X	X				X	X		X		X			X	X		X				C2	
X	X		X		X		X					X	X	X				X			C3	
X	X	X	X	X						X	X	X							X		C4	
			X	X	X	X	X	X	X											X	C5	
								X	X	X	X	X	X	X	X						X	C6

Figure 23.

Figure 25 depicts read check bit generators.

Double bit error detection is accomplished by generating parity on the syndrome bits. Except for the syndrome word of 000000 — no error — even parity will be the result of a double bit error. Hardware implementation is shown in Figure 26. OR-ing the syndrome detects the zero state, which has even parity and prevents flagging this state as a double bit error.

Decoding the syndrome word must be done to invert the one bit in error. Combinational logic will decode only those syndrome states which select the one of sixteen bits for correction. Figure 28 shows the logic of the decoder.

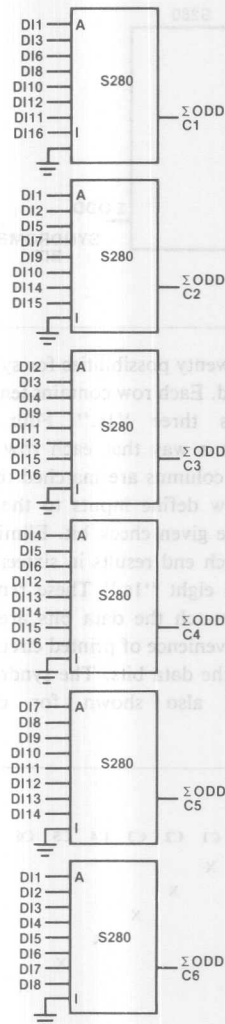


Figure 24. Write Check Bit Generators

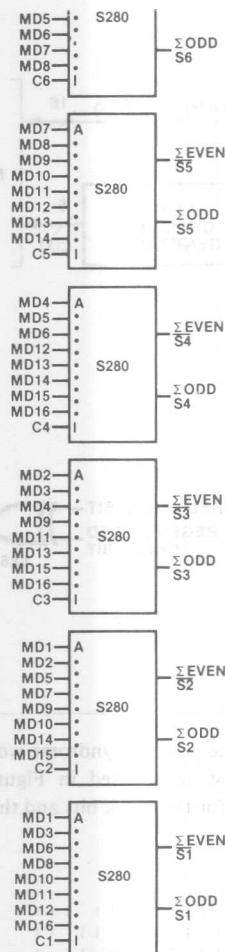


Figure 25. Read Check Bit Generators

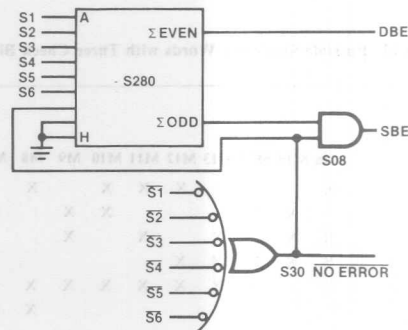


Figure 26. Double Error Decoder



Figure 27. Correction Circuit.

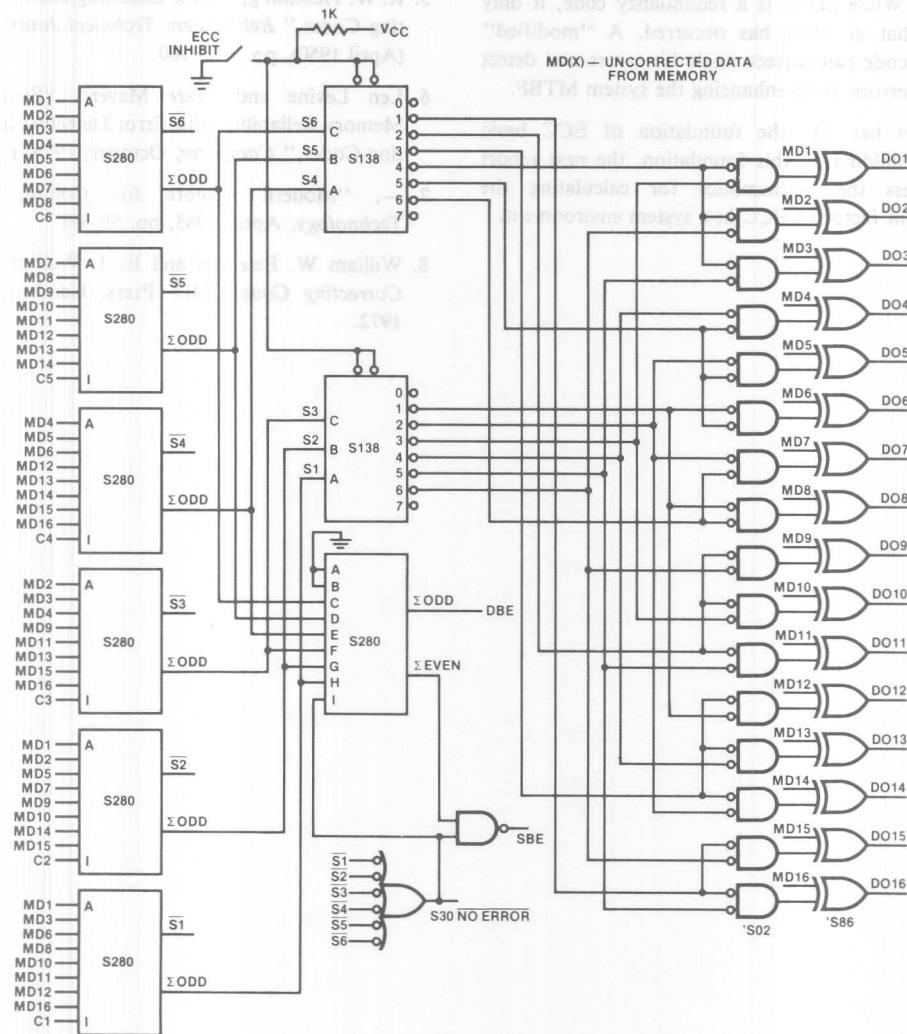


Figure 28. Complete Correction Circuit

Enabling the correction logic, the decoded $B(x)$ signals become "high" to invert the output of the 74S86 exclusive-OR circuits. If the $B(x)$ signals are "low" the output of the correction is the same level as the input. The correction circuit is shown in Figure 29.

Connecting the five circuits as shown in the block diagram of Figure 20 completes the error correction circuitry.

SUMMARY

An unprotected memory has a system MTBF which is approximately equal to the device MTBF divided by the number of devices. Redundancy codes are used to protect memories. While parity is a redundancy code, it only indicates that an error has occurred. A "modified" Hamming code can correct single bit errors and detect double bit errors, truly enhancing the system MTBF.

This report has laid the foundation of ECC basic concepts. Building on this foundation, the next report will address the mathematics for calculating the enhancement factor of ECC in a system environment.

REFERENCES

1. "2107A/2107B N-Channel Silicon Gate MOS 4K RAMs," Reliability Report RR-7, Intel Corporation, September, 1975.
2. "2115/2125 N-Channel Silicon Gate 1K MOS RAMs," Reliability Report RR-14, Intel Corporation, 1976.
3. "2104A N-Channel Silicon Gate 4K Dynamic RAM," Reliability Report RR-15, Intel Corporation, September, 1977.
4. "2116 N-Channel Silicon Gate 16K Dynamic RAM," Reliability Report RR-16, Intel Corporation, August, 1977.
5. R. W. Hamming, "Error Detecting and Error Correcting Codes," *Bell System Technical Journal*, Vol. 26 (April 1950), pp. 147-160.
6. Len Levine and Ware Meyers, "Semiconductor Memory Reliability with Error Detecting and Correcting Codes," *Computer*, October, 1976, pp. 43-50.
7. —, "Modern Algebra for Coding," *Electro-Technology*, April, 1965, pp. 59-66.
8. William W. Peterson and E. J. Weldon Jr., *Error Correcting Codes*, MIT Press, Cambridge, Mass., 1972.

1. INTRODUCTION

This Application Note explains reliability analysis as applied to a typical memory system. (It follows Intel Application Note AP-46, which reviewed basic ECC, **Error Corrections Code**, concepts.) A number of examples demonstrate techniques to calculate reliability of a model memory system, with and without ECC — emphasizing system reliability as a function of the number of devices in a system and the individual device failure rates.

Since a system with ECC can correct a single bit failure and detect double bit errors within an accessed word, it has a decided advantage over a system without ECC. A soft error rate of two or three times device hard failure rate has significantly less effect on the Mean Time Between Failures (MTBF) for a system with error correction. This is quantified as the Enhancement Factor, EF — the ratio of MTBF for two identical systems, one with and one without ECC. The Enhancement Factor can be predicted by the application of statistical analysis.

The general model presented in this Application Note numerically predicts the chance of memory system failures during a specified length of time. It also provides insights into the relationship of device failure mechanisms and soft errors to memory system reliability. Intel® 2117 Dynamic RAM is used in the example memory system. The reliability data for distribution of hard failures was obtained from the 2117 Reliability Report (Intel RR-20).

2. MEMORY CONFIGURATION

2.1 Device

System reliability begins with the smallest physical unit, the memory device. Each device can be considered a system itself, with the smallest functional unit being a single storage cell. Device internal structures have inherent failure mechanisms affecting individual memory cells.

The structure of a typical RAM device consists of two-dimensional coordinate-addressed arrays of memory cells arranged in rows and columns, such as the Intel® 2117 Dynamic RAM shown in Figure 2. This device contains 16384 cells arranged in a 128 row by 128 column matrix; each cell is selected by an encoded 7-bit row and 7-bit column address.

2.2 System

An array of memory devices on one or more circuit boards forms a typical memory system. A system is defined by n bits per word, x words per

page and p pages per system. Note that a "page" is defined as the number of memory words formed by a minimum set of memory components.

For example, 16K by 1 RAMs would have a minimum page size of 16384 words.

Figure 1 represents such a system, with the horizontal axis corresponding to parallel, address-accessed data bits and the vertical axis corresponding to the series stacking of words and pages. This memory structure is used for the model system.

3. ERROR CLASSIFICATION

The 2117 failure mechanisms illustrated in Figure 3 are fairly representative for today's RAM devices. These can be categorized as **hard failures** and **soft errors**.

3.1 Hard Failures

Hard failures are permanent physical defects, such as shorts, open leads, micro-cracks or other intrinsic flaws. They are classified as single cell failures, row failures, column failures, combined row-column failures, half-chip failures and full-chip failures.

The failure type distribution within a device is a function of the device design. Typical ratios are 50% single cell failures, 40% row or column failures, 10% combined failures and less than 0.1% half-chip or full-chip failures. (Refer to Figure 4.) The accumulative independent events are expressed as a single numeric value for the combined failure rate of the device (EQ:1a). The standard mathematical symbol for device failure rate is the Greek letter Lambda, λ ; i.e., $\lambda = 0.027\%/1000 \text{ hrs}$.

$$\text{EQ:1a } \lambda_{\text{hrd}} = \lambda_{\text{single}} + \lambda_{\text{row}} + \lambda_{\text{column}} + \lambda_{\text{row/col}} + \lambda_{\text{halfchip}} + \lambda_{\text{fullchip}}$$

3.2 Soft Errors

In contrast to hard failures, soft errors are characterized as being random in nature, non-recurring, non-destructive single cell errors.

Traditional soft errors are caused by noisy system environments, poor system design, or rare combinations of noise, data patterns, and temperature effects which push the RAM beyond its normal specified range of operation. This type of soft error has not been included in the analysis to follow because it is associated with system level problems and the rate of failure is difficult to quantify; in any case it is assumed to be quite small.

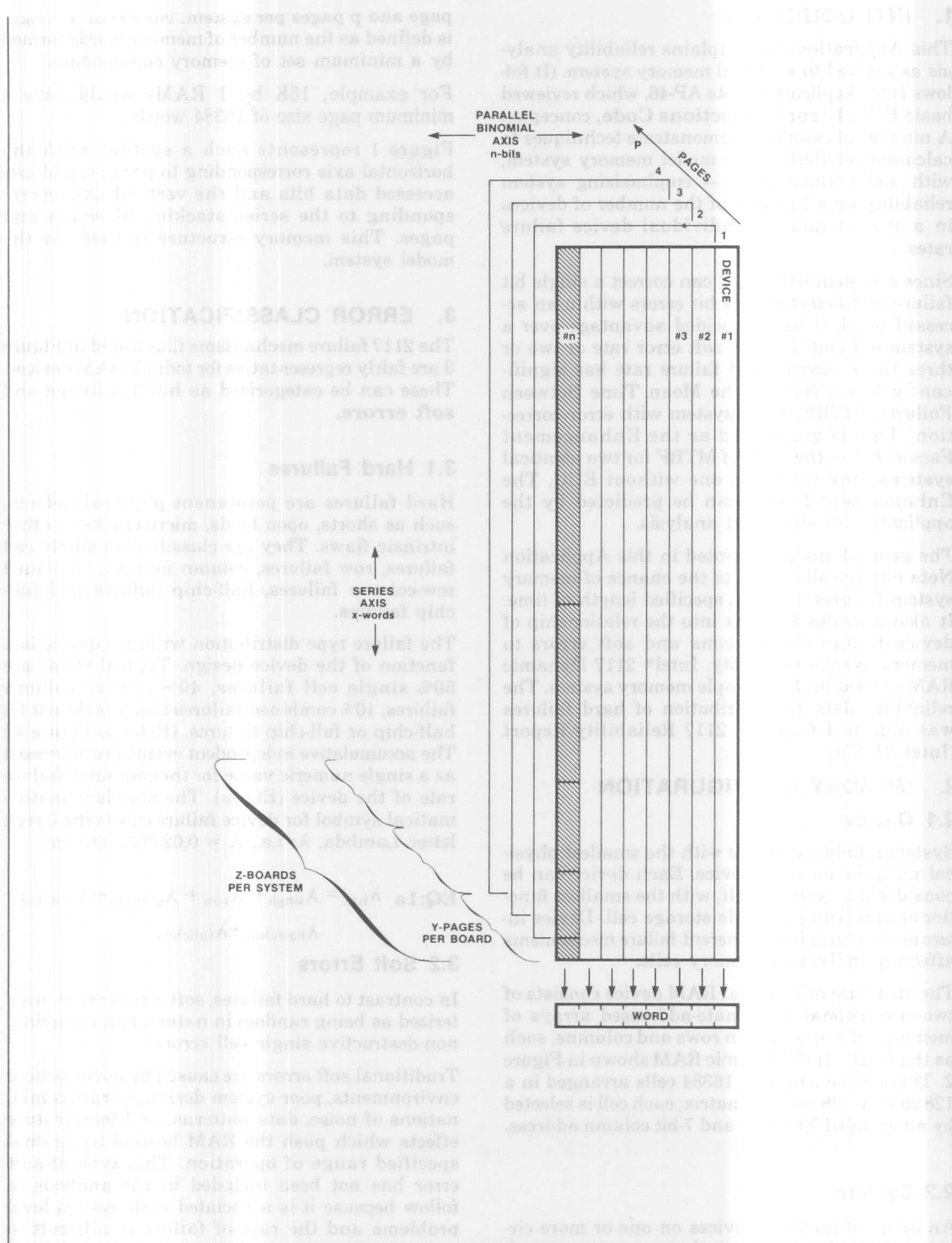
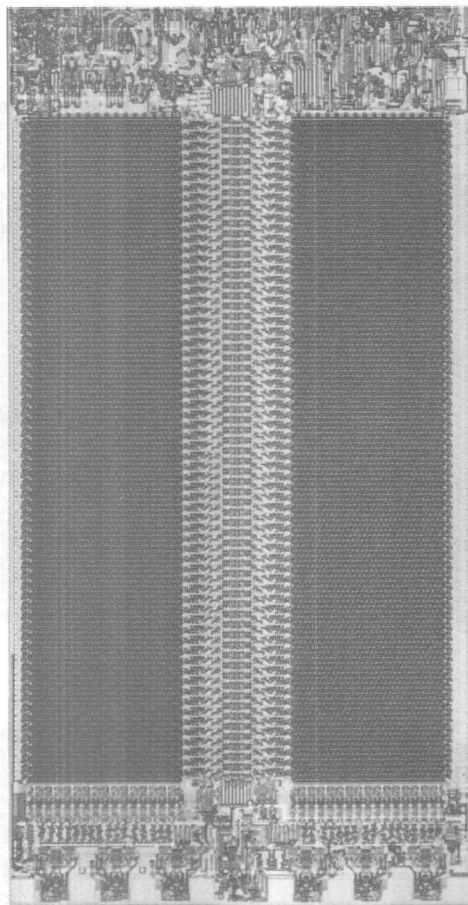


Figure 1. Memory Configuration



EXAMPLE: 2117
128 COLUMNS BY 128 ROWS

Figure 2. Random Access Memory Device

Other soft errors are caused by ionizing radiation of alpha particles changing memory cell charge in semiconductor substrates with high impedance nodes. The data bit error is realized during a memory read to the failing cell. These errors are purged by rewriting (restoring) the correct data bit information to the cell. The failure rate for this type of soft error is stated separately from hard failures because of its unique properties.

The total device failure rate becomes:

$$\text{EQ:1b } \lambda_{\text{dev}} = \lambda_{\text{hrd}} + \lambda_{\text{sft}}$$

The pie graph in Figure 5 depicts the combined distribution of both hard and soft errors.

defined as "the probability that a component will operate within specified limits, for a given period of time"¹. The definition includes the term "probability", a quantitative measure for chance or likelihood of occurrence, of a particular form of event — in this case, operation without failure within specified limits. In addition to the probabilistic aspect, the reliability definition also involves length of operational time.

Since reliability is concerned with events which occur in the time domain, they are classified as incidental failures, which do not cluster around any mean life period, but occur at random time intervals. The exact time of failure cannot be predicted; however, the probability of occurrence or non-occurrence of a statistical mean in a given operating frame of time can be analyzed by the theories of probability. Since exact formulae exist for predicting the frequency of occurrence of events following various statistical distributions, the chance or probability of specified events can be derived.

4.1 Component Reliability

Memory systems are operated where failures occur randomly due only to chance causes. The fundamental principles of reliability engineering predict the failure rate of a group of devices which will follow the so-called bathtub curve in Figure 6. The curve is divided into three regions: Infant Mortality, Random Failures, and Wearout Failures. All classes of failure mechanisms can be assigned to these regions.

Infant Mortality, as the name implies, represents the early life failures of a device. These failures are usually associated with one or more manufacturing defects. Memory device failures occurring as the result of Infant Mortality have been eliminated by corrective actions relating design, inspection, and test methods.

Wearout failures occur at the end of the device's useful life and are characterized by a rising failure rate with time as the device's "wearout" both physically and electrically. This does not occur for hundreds of years for integrated circuits.

The Random Failure portion of the curve represents the useful period of device life. As stated, memory devices are operated in systems during this period when failures occur randomly. The number of failures occurring during any time interval within the "Random" period is related only to the total number of memory components

¹ Reliability Mathematics — Amstadter

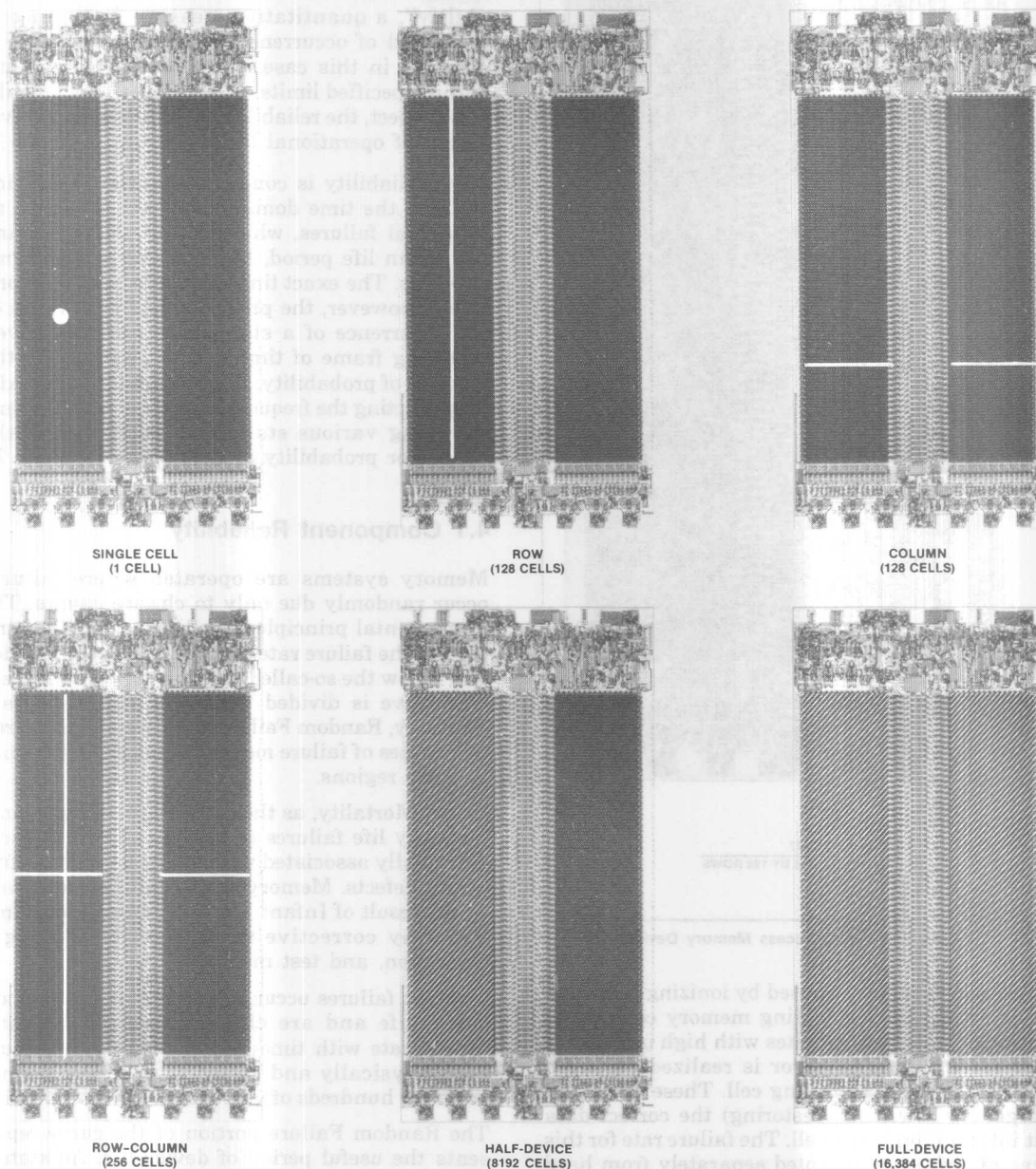


Figure 3. Failure Geometry — 2117 Example

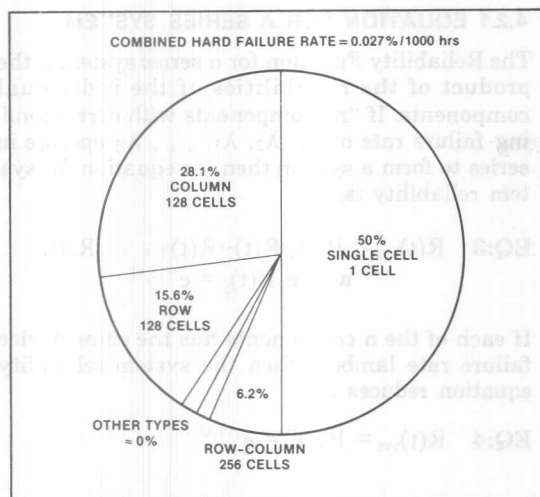


Figure 4. Failure Distribution — 2117 Example

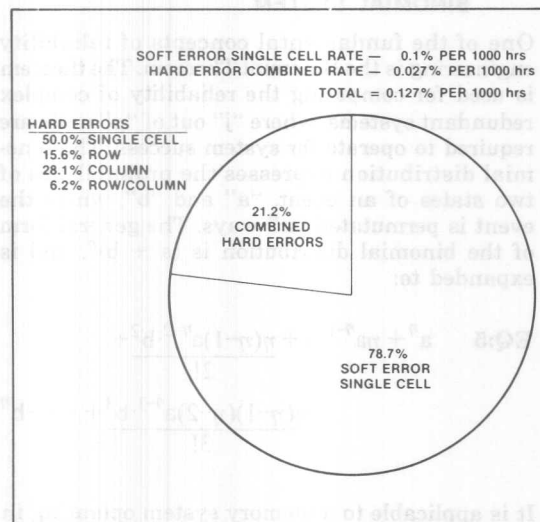


Figure 5. Combined Distribution of Failure Type

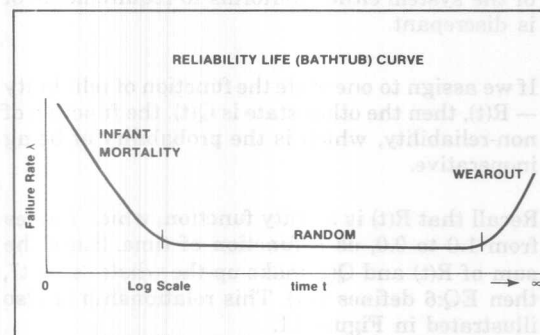


Figure 6. Reliability Life Curve

operating. If sufficient numbers are operated, and the measured interval is long enough, failure rate approaches some relative constant value. For any given component type, the failure rate value will depend on operating and external environmental conditions (voltage, temperature, timing, etc.) and will be characteristic of this set of conditions. When the conditions change, the failure rate will correspondingly change.

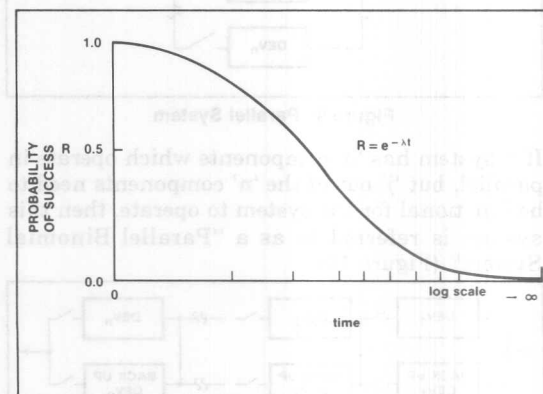
For example, if 500 devices are tested for 1,500 hours and two failures were observed during the test interval, then the failure rate is two failures per 750,000 device-hours or one failure per 375,000 device-hours. For commonality, device failure rates are expressed as a percentage value per 1000 device-hours. The above example then becomes .00266 failures per 1000 device-hours or $\lambda_{dev} = 0.27\%$ per 1000 hours. This is an overly simplified statement on determining the device failure rate. Many tests, designed to stress the devices over operating conditions and margins, are used in the final analysis for the specification of device failure rates.

4.1.1 RELIABILITY FUNCTION $R(t)$

The Reliability Function, $R(t)$, follows an inverse, natural logarithmic curve, which expresses the rate of change for a memory component from an operational state to a failure or error condition. The curve is a familiar one to the physical scientists because of its relationship to growth and decay.

The general function for reliability is given in EQ:2 where the exponent ($\lambda \cdot t$) represents the device failure "lambda" times the independent time variable "t". The graph in Figure 7 shows the shape of the R-function curve.

$$\text{EQ:2} \quad R(t) = e^{-\lambda t}$$

Figure 7. $R(t)$ - Reliability Function

$$R(0) = 1.0 \text{ and } R(\infty) = 0.0$$

The distribution is a one-parameter type; in that once the failure rate is established, the reliability function is completely defined. For high or low failure rates the general shape of the curve remains the same, but is adjusted along the time axis.

4.2 System Reliability

Just as there is a functional relationship between the components and the system, there is a functional relationship between component reliability and system reliability. If a failure in any one of the components of a system causes the entire system to fail, the system is a "Series System" (Figure 8).

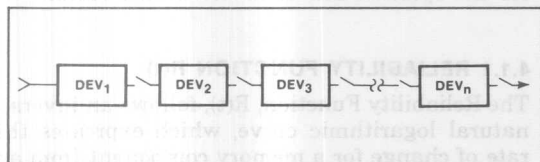


Figure 8. System of Series Components

If all the component devices must fail before the system fails, the system is a "Parallel System" (Figure 9).

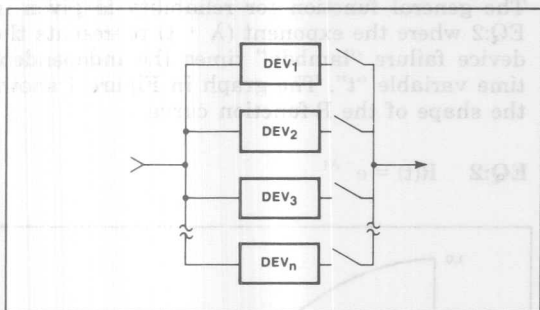


Figure 9. Parallel System

If a system has 'n' components which operate in parallel, but 'j' out of the 'n' components need to be functional for the system to operate, then this system is referred to as a "Parallel Binomial System" (Figure 10).

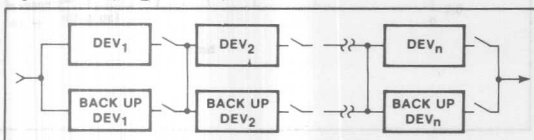


Figure 10. Parallel Binomial System

product of the reliabilities of the individual components. If "n" components with corresponding failure rate of $\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_n$ operate in series to form a system then the equation for system reliability is:

$$\text{EQ:3 } R(t)_{\text{sys}} = R(t)_1 \cdot R(t)_2 \cdot R(t)_3 \cdot \dots \cdot R(t)_n$$

where $R(t)_i = e^{-\lambda_i \cdot t}$

If each of the n components has the same device failure rate lambda, then the system reliability equation reduces to:

$$\text{EQ:4 } R(t)_{\text{sys}} = R(t)^n = e^{-n\lambda t}$$

4.2.2 EQUATION FOR A PARALLEL BINOMIAL SYSTEM

One of the fundamental concepts of reliability engineering is the Binomial Theorem. The theorem is used for computing the reliability of complex redundant systems, where "j" out of "n" units are required to operate for system success. The binomial distribution expresses the probabilities of two states of an event, "a" and "b", where the event is permuted "n" ways. The general form of the binomial distribution is $(a + b)^n$, and is expanded to:

$$\text{EQ:5 } a^n + \eta a^{n-1} \cdot b + \frac{\eta(\eta-1)}{2!} a^{n-2} \cdot b^2 + \frac{\eta(\eta-1)(\eta-2)}{3!} a^{n-3} \cdot b^3 + \dots + b^n$$

It is applicable to a memory system operating in parallel; i.e., when there are only two possible states or results of an event — when a component of the system either conforms to requirements or is discrepant.

If we assign to one state the function of reliability — $R(t)$, then the other state is $Q(t)$, the function of non-reliability, which is the probability of being inoperative.

Recall that $R(t)$ is a unity function, which ranges from 1.0 to 0.0, as a function of time. Since the sum of $R(t)$ and $Q(t)$ make up the whole "event", then EQ:6 defines $Q(t)$. This relationship is also illustrated in Figure 11.

$$\text{EQ:6 } R(t) + Q(t) = 1, \text{ then } Q(t) = 1 - R(t)$$

By substituting $R(t)$ and $Q(t)$ respectively for a and b , where $R(t)$ is the probability of a device being good, $Q(t)$ is the probability of the same device being defective, and "n" the number of units in parallel, then:

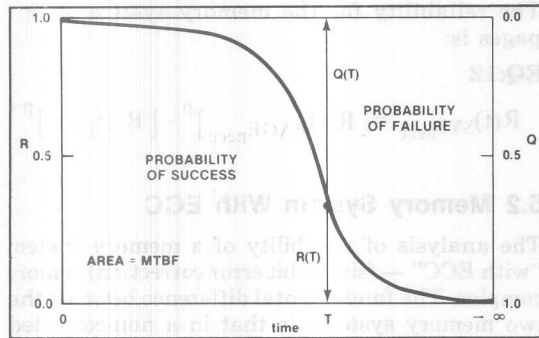


Figure 11. $Q(t) = 1 - e^{-\lambda t}$

$$\text{EQ:7 } [R + Q]^n = 1$$

Note, for simplicity, all references to (t) for the reliability and non-reliability functions will not be indicated, but implied.

It follows that the expansion of $[R + Q]^n$ must also equal unity example

$$\text{EQ:8 } R^n + nR^{n-1} \cdot Q + \frac{n(n-1)R^{n-2} \cdot Q^2}{2!} + \frac{n(n-1)(n-2)R^{n-3} \cdot Q^3}{3!} + \dots + Q^n = 1$$

We can next examine the meaning of each term in the series on the left side of EQ:8. Suppose that there are "n" identical components of a system, of which the probability of a component being operative is R, and that the probability of its being inoperative is Q or $(1 - R)$. If there is only one component ($n = 1$), then the probability of its being not defective is simply R.

If there are two components ($n = 2$), then the probability of both being operative is $R \times R = R^2$; and if there were three components, then the probability of all three being good is R^3 . Consequently, if there are "n" components, the chance of all "n" units being operative is R^n and the first term in the series R^n is the probability of all components being operational.

Next, suppose there are two components X and Y, one is operative and one has failed. There are two ways that this can occur: X is operational and Y fails, with the probability $R_x \cdot Q_y$; or X fails and Y is operational, with the probability $Q_x \cdot R_y$. Since these are mutually exclusive and constitute all possible combinations of one operative component and one failure, the total probability is $(R_x Q_y) + (Q_x R_y)$, or $2RQ$.

If there are three components X, Y, and Z, of which two are operative and one fails, then three possible combinations exist: X and Y are operational and Z fails, X and Z operational and Y fails, and Y and Z operational and X fails. The probability of each combination is $(R_x R_y Q_z) + (R_x Q_y R_z) + (Q_x R_y R_z)$.

Again, since each combination is mutually exclusive and together they constitute all possible combinations, the probability of two operational devices and one failure is $3R^2 \cdot Q$. Similarly, if there are n component-devices, the probability of all but one being operative is $nR^{n-1} \cdot Q$. Thus, the second term of the binomial expansion series is the probability of exactly one device failure, and all other devices being good.

By extending these derivations to cover each succeeding term, we find that the third term is the probability of exactly two failed components, the fourth term is the probability of exactly three failures and so on. There are $n + 1$ terms in the expansion, and the last term Q is the probability all components are inoperative.

The reliability of a group of redundant items depends not only on the reliability of each individual item and on the number of items in redundant configuration, but also on how many are required to operate to achieve system success. If all are required, then the first term of the binomial series represents system success. In this case there is really no redundancy. However, if all but one are required (one failure permitted), then success is achieved if no failures occur or exactly one failure occurs within word accessed from a page of memory. The system reliability is then the sum of the first two terms of the series.

If two failures are permitted, then the sum of the first three terms represents the probability of system success. In general, if r failures are permitted, system success is the sum of the first r + 1 terms.

The general equation then for a binomial system, permitting one error, which is representative of a memory system with single bit error correction — ECC per accessed word is expressed as:

$$\text{EQ:9 } R_T(t) = \underbrace{R^n}_{1\text{st}} + \underbrace{nR^{n-1} \cdot Q}_{2\text{nd}} - \text{binomial terms}$$

Note that the remaining terms of the binomial expansion represent all combinations of failures that are greater than one failure, up to and including all components failing. $R_T(t)$ is still a unity function of reliability and has a converse $Q_T(t)$, where $Q_T(t) = 1 - R_T(t)$. Thus, Q_T represents the 3rd through n-th terms of the binomial.

5. RELIABILITY ANALYSIS USING PAGE/SYSTEM APPROACH

The analysis of the model system in Figure 1 begins with EQ:2 at the smallest non-redundant failure level; by using standard rules for series and parallel reliability, the combination of these device exponential expressions will yield the system reliability equation. The method of approach will be to calculate the reliability of a page of memory and treat subsequent pages as a series system where:

$$\text{EQ:10 } R(t)_{\text{system}} = [R(t)_{\text{page}}]^P$$

For clarity, the reliability of power supplies, fans, backplane connections, TTL support logic, etc. will not be included. These items can be merged in the final analysis by the reader as additional series system equations for each type.

5.1 Memory System Without ECC

The analysis of reliability of a memory system "without" any form of ECC is simply the first term of the binomial equation EQ:9. Since this term represents reliability of all components in a page of memory without redundancy, it is equivalent to a "series system" equation (EQ:4). Therefore, the equation for a page of memory without ECC is:

$$\text{EQ:11 } R(t)_{\text{PAGE}_{\text{nec}}} = R(t)_{\text{DEV}_{\text{nec}}}^{\eta} = e^{-\lambda \cdot n \cdot t}$$

where "n" is the number of components in the page and λ_{dev} is the device combined failure rate

The reliability for the memory system of "i" pages is:

EQ:12

$$R(t)_{\text{SYS}_{\text{nec}}} = [R(t)_{\text{PAGE}_{\text{nec}}}]^P = [R(t)_{\text{DEV}}]^{P \cdot \eta}$$

5.2 Memory System With ECC

The analysis of reliability of a memory system "with ECC" — (single bit error correction) is more complex. The fundamental difference between the two memory systems is that in a non-corrected system, any error — no matter the type, single cell failure, row failure, soft error, etc. — is considered a system failure. In a memory system with ECC, a system level failure only occurs when more than one bit has failed in an accessed word.

Thus in the analysis of a System with ECC, we must deal with the probabilities of each failure type occurring in random combinations which align within a word of memory to cause multiple bit failures as shown in Figure 12.

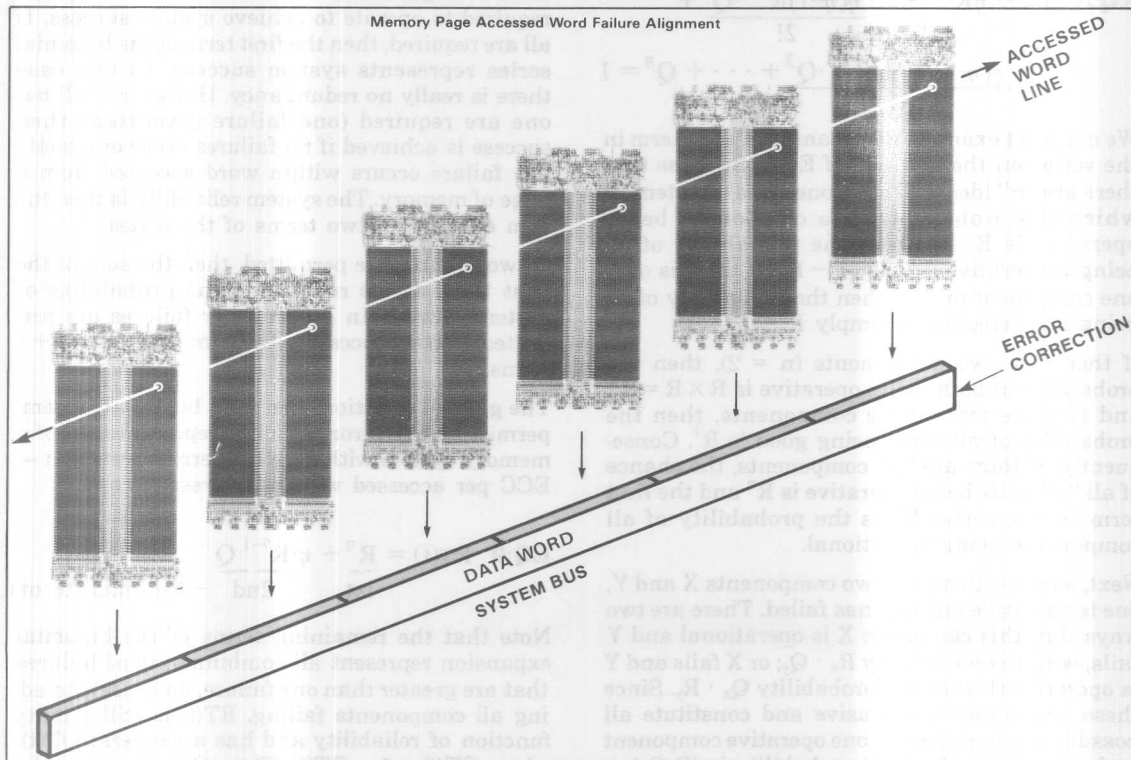


Figure 12. Memory Page Accessed Word Failure Alignment


$$R(t)_{\text{PAGEcc}} = [R(t)_x^\eta + \eta \cdot R(t)_x^{\eta-1} \cdot Q(t)_x]^{\ell_x}$$

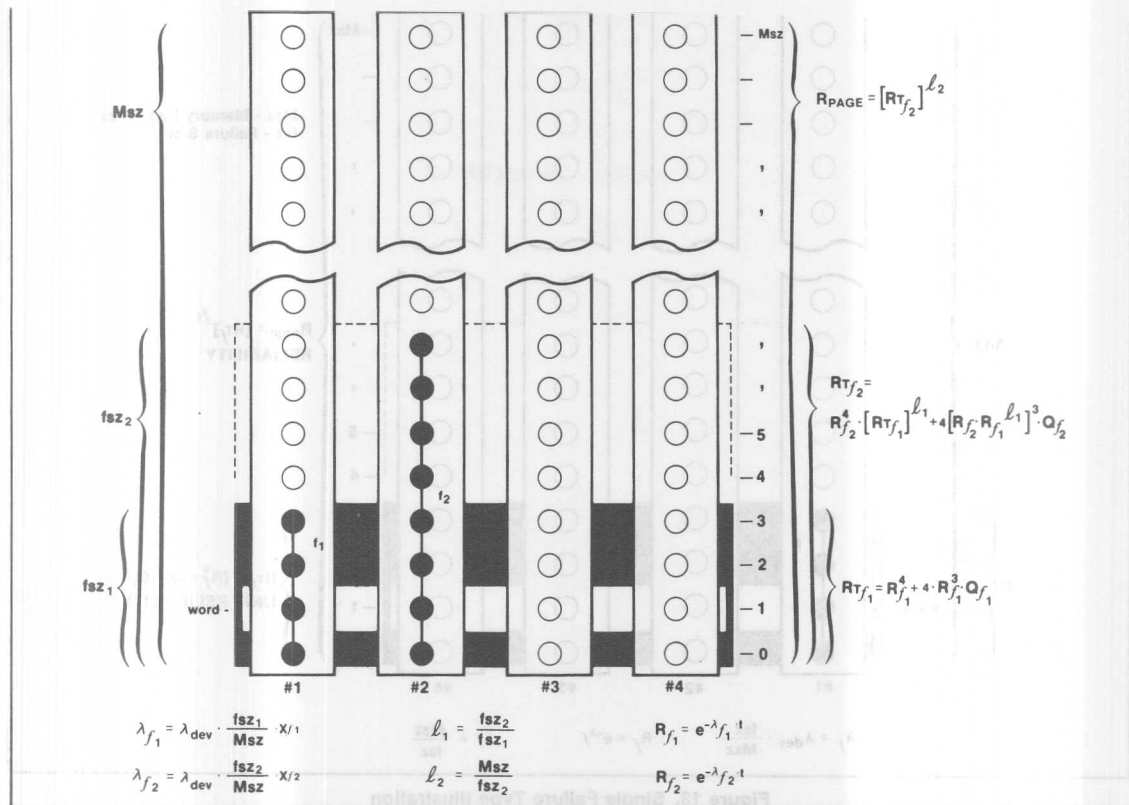


Figure 14. Multiple Failure Type Illustration

Now that the binomial equation technique has been applied to a single failure type, let's expand the process to cover more than one failure type. By the process of combining or permutating these failure types, the Reliability Function can be calculated. Figure 14 shows a four component system with the probability that two failure types f_1 or f_2 can occur in each component. Both failure types affect fsz_1 and fsz_2 number of cells during a failure, respectively. The calculation begins with evaluating the probability of f_1 occurring (EQ:14a) and merging by a second calculation the probability of failure type f_2 . (EQ:14b).

$$\text{EQ:14a } RT_1 = R_{f_1} + \eta R_{f_1}^{\eta-1} \cdot Q_{f_1}$$

$$\text{EQ:14b } RT_2 = R_{f_2} \cdot [RT_1^{\ell_2}] + \eta [R_{f_2} \cdot R_{f_1}^{\ell_2}]^{\eta-1} \cdot Q_{f_2}$$

NOTE: with λ_{dev} representing more than one failure type, f_1 and f_2 , λ_{dev} must be proportioned to the "failure-type-distribution" in determining the unit failure rates λ_{f_1} and λ_{f_2} . The term X_{f_1} and X_{f_2} are introduced to quantify the failure type distribution as a percentage. (Ref: EQ:1 and Figure 5).

EQ:14c is the unit failure rate equation for f_1 and f_2 in this case.

EQ:14c

$$\lambda_{f_1} = \lambda_{\text{dev}} \cdot X_{f_1} \cdot \frac{\text{fsz}_{f_1}}{\text{Msz}} \quad \lambda_{f_2} = \lambda_{\text{dev}} \cdot X_{f_2} \cdot \frac{\text{fsz}_{f_2}}{\text{Msz}}$$

The total reliability for the page in Figure 13b is given by equation 14d.

$$\text{EQ:14d } R(t)_{\text{page}} = [RT_2]^{\frac{\text{Msz}}{\text{fsz}_2}}$$

By expanding on this process the equation for a system of memory components with these failure types: f_1, f_2, f_3 is given in EQ:15.

EQ:15

$$RT_3 = R_{f_3} \cdot [RT_2]^{\ell_3} + \eta [R_{f_3} (R_{f_2} (R_{f_1})^{\ell_2})^{\ell_3}]^{\eta-1} \cdot Q_{f_3}$$

We can now formulate a general set of equations for multiple (f_i) failure types in an error corrected system.

The full model under analysis in this report has six failure types, as described in the section on Error Classification. The reliability calculations for a page of memory must permute all combinations of these six failure types. It is accomplished by the set of equations in EQ:16.

EQ:16

$$\begin{aligned}
 & \left[i \leftarrow \begin{array}{l} N \\ \ell_i = \frac{fsz_i}{fsz_{i-1}} \\ \lambda_{f_i} = \lambda_{dev} \cdot X_i \cdot \frac{fsz_i}{MsZ} \end{array} \right] \\
 & R(t)_{PAGE_{ecc}} = \left\{ i \leftarrow \begin{array}{l} N \\ R_i = e^{-\lambda_{f_i} \cdot t}, Q_i = 1 - R_i \\ RS_i = R_i \cdot (RS_{i-1})^{\ell_i} \\ RT_i = R_i^{\eta_i} \cdot (RT_{i-1})^{\ell_i} + \eta(RS_i)^{\eta-1} \cdot Q_i \end{array} \right\} \left\{ \frac{MsZ}{fsz_N} \right\} \\
 & R(t)_{SYSTEM_{ecc}} = [R(t)_{PAGE_{ecc}}]^{Pages} \\
 & \text{restrictions: } RS_0 = RT_0 = fsz_0 = 1.
 \end{aligned}$$

The process begins at the word level with soft errors and gradually increases the area of evaluation to single cell hard failures, then row or column failures, combined row/column failures, half-chip failures, and finally full-chip failures.

Illustrated in Figures 15 and 16 are the six iterative steps to merge all combinations of failure types — $f_1, f_2, f_3, f_4, f_5, f_6$.

The first step calculates the chance of a single word of the memory page not having more than one soft error.

The second step calculates the probability of not having more than one single-cell hard failure and merges step #1, for a combined result that no more than one failure caused by either soft error or single-cell failure has occurred within the single word analyzed.

The third step calculates for row failures and merges with step #2 all combinations of the three failure types. Using the 2117 example memory system from Figure 6 to illustrate this point — a row or column failure affects 128 memory words — the combined result from step #2, which analyzed a single word, is raised by the exponent 128 as a series equation. The combined result for step #3 is the probability of not having a system failure due to any of the failure types f_1, f_2, f_3 , in any given word for a 128-word block.

This process continues up to step six, which is the calculation for all six failure types occurring in all combinations that would cause a system failure within the page of memory. The analysis of each step therefore raises the results of each previous step by the exponent ℓ_i .

5.2.2.1 Mean Time Between Failures

The Mean Time Between Failures (MTBF) for a memory system, with or without ECC, is given in EQ:17. MTBF is calculated by integrating the system reliability function, $R(t)_{sys}$, from $t = 0$ to infinity.

EQ:17
$$MTBF_{sys} = \int_0^{\infty} R(t)_{sys} \cdot dt$$

On the average a system will fail once every $MTBF_{sys}$ hours. The relationship between MTBF and the R function is shown in Figure 17.

The bottom line conclusions on the effect that error-correction has on a given memory system is calculated by comparing the resultant $MTBF_{sys-ecc}$ projection with the $MTBF_{sys-necc}$ of a similar system without ECC. The improvement of a memory system with error correction logic over a comparable system without is expressed by EQ:18 as the enhancement factor EF.

EQ:18
$$EF = \frac{MTBF_{sys-ecc}}{MTBF_{sys-necc}}$$

5.2.2.2 Mean Time To Failures

The Mean Time To Failure (MTTF) is similar in concept to MTBF, but differs in that it represents the effects of maintenance on an error corrected memory system. When a maintenance policy is adopted which allows for the replacement of failed components before the system fails, system failure is postponed (depending on how often the system is inspected and maintained). With this policy a memory system fails less frequently than it does without maintenance; it is assured that every new operating period after inspection starts with full redundancy restored. The maintained system Mean Time To Failure thus becomes greater than $MTBF_{sys}$.

If preventive maintenance is performed at an arbitrary time T, then EQ:19 expresses mean time to failure.

EQ:19
$$MTTF = \frac{\int_0^T R(t)_{sys-ecc} \cdot dt}{1 - R(T)_{sys-ecc}}$$

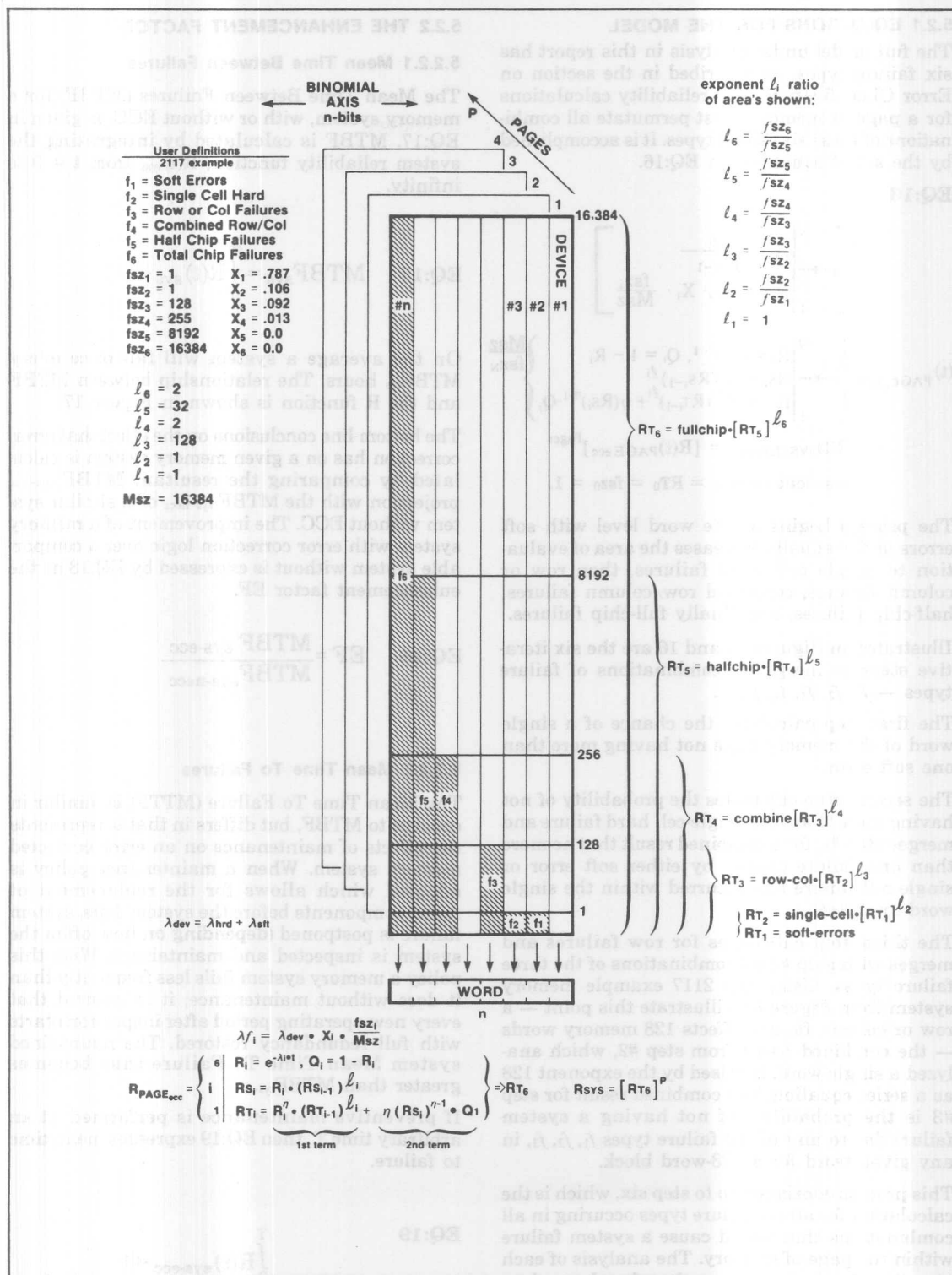


Figure 15.

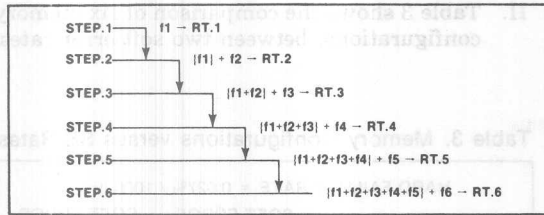


Figure 16.

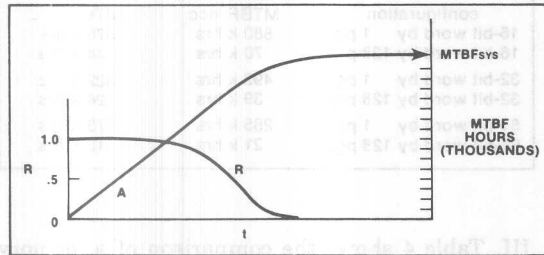


Figure 17.

Figures 18 and 19 show the relationship of MTTF to the R function and MTTF to MTBF respectively.

The enhancement of a memory system with maintenance over a comparable system without ECC is expressed in EQ:20.

$$\text{EQ:20} \quad EF_{\text{mnt}} = \frac{\text{MTTF}}{\text{MTBF}_{\text{sys-ecc}}}$$

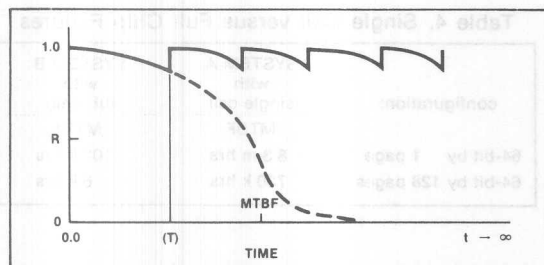


Figure 18.

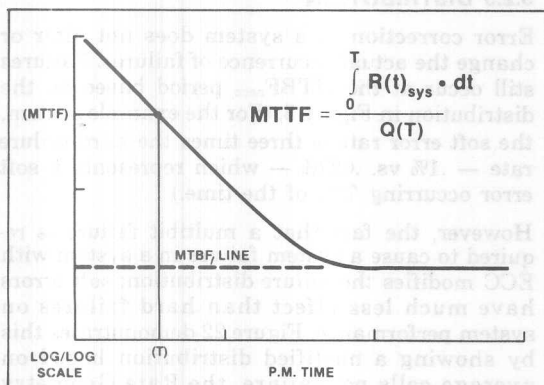


Figure 19.

5.2.3 SOFT ERROR SCRUBBING

In the previous sections on MTBF and MTTF, soft errors and hard errors were treated the same. They both accumulated to cause system failure or were removed at scheduled preventive maintenance (PM) intervals.

However, soft errors can have their own special maintenance function. Recall that soft errors can be purged from a system with ECC by rewriting (restoring) the correct data bit information to the failing memory cell. (Provided that no other bit within the word containing the soft error has failed.) Thus it is possible for the system to maintain itself by software, etc. This special maintenance function of scrubbing soft errors at predetermined intervals is incorporated into the system reliability equations by merely resetting the time parameter t for the soft error portion of the equations.

Figure 20 shows the relationship of soft error scrubbing on MTBF and the system R functions.

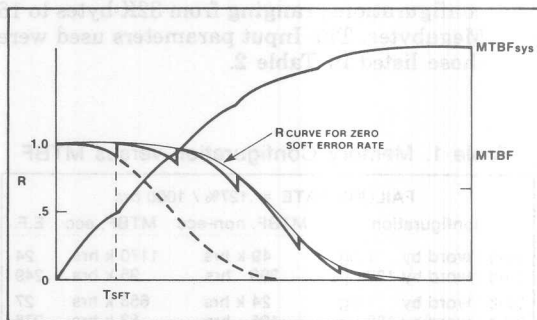


Figure 20.

5.2.4 APPLYING THE MODEL EQUATIONS

The basic set of equations for a model are derived from EQ:16. The application of these equations is best suited for implementation on a computer. An example computer program is available on request.

Figure 21 illustrates a simplified block diagram of the model.

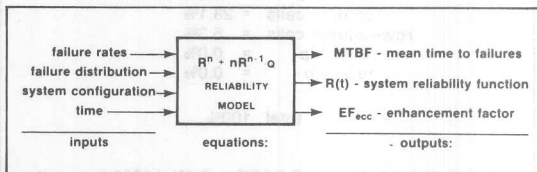


Figure 21.

The required user inputs are for component parameters — total memory size, number of rows and columns, hard failure rate, soft error rate, and

failure mode distribution; for system parameters — memory word size, ECC check bits, number of pages, interval of time, and soft error scrub time.

Output is a set of discrete values of the reliability function representing the complete memory system as a function of time.

The integral functions for MTTF and MTBF are evaluated by the trapezoidal rule of integration.

EQ:21

$$MTBF = \sum_{i=1}^{\infty} \frac{1}{2} [R_{sys,i-1} + R_{sys,i}] \cdot \Delta Time$$

where $R_{sys,0} = 1$

Based on the Intel® 2117 Dynamic Ram, the following three sections — (I, II, III) — compare various system configurations and failure rate parameters.

- I. Table 1 shows the comparison of six memory configurations, ranging from 32K-bytes to 16 Megabytes. The Input parameters used were those listed in Table 2.

Table 1. Memory Configuration versus MTBF

FAILURE RATE = .127% / 1000 hrs			
configuration	MTBF, non-ecc	MTBF, ecc	E.F.
16-bit word by 1 pg	49 k hrs	1170 k hrs	24
16-bit word by 128 pgs	390 hrs	95 k hrs	249
32-bit word by 1 pg	24 k hrs	658 k hrs	27
32-bit word by 128 pgs	195 hrs	53 k hrs	278
64-bit word by 1 pg	12 k hrs	355 k hrs	29
64-bit word by 128 pgs	98 hrs	29 k hrs	299

Table 2. Model Input Parameters

Combined HARD FAILURE RATE = 0.027% / 1000 hours	
Failure distributions:	
single cell	= 50.0%
row cells	= 15.6%
column cells	= 28.1%
row-column cells	= 6.3%
half-chip	= 0.0%
full-chip	= 0.0%
total	100%
SOFT ERROR FAILURE RATE = 0.1% / 1000 hrs - est.	

These results show an enhancement factor of approximately 27 for a single page of memory and over 278 for 128 pages.

- II. Table 3 shows the comparison of six memory configurations, between two soft error rates.

Table 3. Memory Configurations versus SE Rates

HARD FAILURE RATE = 0.027% / 1000 hrs		
configuration	SOFT ERROR RATE	SOFT ERROR RATE
	.2% / 1000 hrs	.5% / 1000 hrs
16-bit word by 1 pg	MTBF, ecc 880 k hrs	MTBF, ecc 575 k hrs
16-bit word by 128 pgs	70 k hrs	44 k hrs
32-bit word by 1 pg	492 k hrs	322 k hrs
32-bit word by 128 pgs	39 k hrs	24 k hrs
64-bit word by 1 pg	265 k hrs	173 k hrs
64-bit word by 128 pgs	21 k hrs	13 k hrs

- III. Table 4 shows the comparison of a memory device with one failure type. The failure types compared are devices with a single cell failure modes and full-chip failure modes.

System A has devices with only "single cell" failure types and System B has only "full-chip" type. All other parameters are identical. Both system failure rates are 0.027%/1000 hrs.

Table 4. Single Cell versus Full Chip Failures

configuration:	SYSTEM A with single cell	SYSTEM B with full-chip
	MTBF	MTBF
64-bit by 1 page	8.3 m hrs	103 k hrs
64-bit by 128 pages	730 k hrs	6 k hrs

5.2.5 DISTRIBUTION

Error correction in a system does not alter or change the actual occurrence of failures. Failures still occur at the $MTBF_{nec}$ period based on the distribution in Figure 5. (For the example system, the soft error rate is three times the hard failure rate — .1% vs. .027% — which represents a soft error occurring 78% of the time.)

However, the fact that a multibit failure is required to cause a system failure in a system with ECC modifies the failure distribution; soft errors have much less effect than hard failures on system performance. Figure 22 demonstrates this by showing a modified distribution based on average cells per failure, the Rate Geometry Product, RGP.

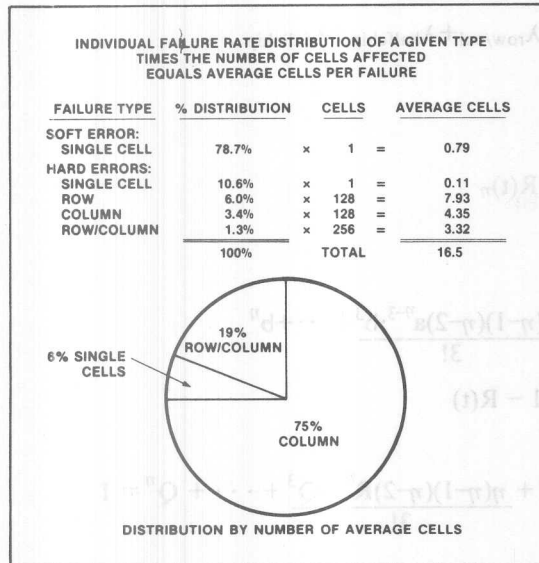


Figure 22.

The illustration shows the statistical average cell failure for each type derived by taking the product of the component failure rate distribution times the number of cells affected. For the 2117 example device, the total average cell failure is 16.2 of which 11.8 are column and row failures.

Intuitively, it can be seen that row and column failures are the most predominant, while the least predominant are soft errors and single cell hard errors.

6. SUMMARY

This Application Note presents step-by-step procedures for calculating system reliability. In a system without ECC, a fault of any type can cause system failure — predominantly types with the highest failure rates. In a system with ECC, only multi-bit errors within the same word cause system failure — predominantly types with the highest average cell errors as defined by the Rate Geometry Product. An Enhancement Factor, comparing a system without ECC to one with ECC, can be used to determine if error correcting techniques are advantageous for any specific memory system.

References

1. Randall C. Cork, "Reliability with Error-Detecting and Correcting Codes in Semiconductor Memories," Ph.D. dissertation, Arizona State University, 1975.
2. Bertram L. Amstadter, Reliability Mathematics Fundamentals; Practices; Procedures, McGraw Hill, N.Y., 1971.
3. Byron L. Newton, Statistics for Business, Science Research Associates, 1973.
4. Carl-Erik W. Sundberg, member IEEE, "Erasure and Error Decoding for Semiconductor Memories", IEEE 1978.
5. Peter Elias, "Error Free Coding", MIT.
6. S.K. Wang & K. Lovelace, "Improvement of Memory Reliability by Single-Bit-Error Correction", Texas Instruments Inc.

AP-73

$$\text{EQ:1b} \quad \lambda_{\text{dev}} = \lambda_{\text{hrd}} + \lambda_{\text{sft}}$$

$$\text{EQ:2} \quad R(t) = e^{-\lambda t}$$

$$\text{EQ:3} \quad R(t)_{\text{sys}} = R(t)_1 \cdot R(t)_2 \cdot R(t)_3 \cdot \dots \cdot R(t)_\eta$$

where $R(t)_i = e^{-\lambda_i \cdot t}$

$$\text{EQ:4} \quad R(t)_{\text{sys}} = R(t)^\eta = e^{-\eta \lambda t}$$

$$\text{EQ:5} \quad a^\eta + \eta a^{\eta-1} \cdot b + \frac{\eta(\eta-1)a^{\eta-2} \cdot b^2}{2!} + \frac{\eta(\eta-1)(\eta-2)a^{\eta-3} \cdot b^3}{3!} + \dots + b^\eta$$

$$\text{EQ:6} \quad R(t) + Q(t) = 1, \text{ then } Q(t) = 1 - R(t)$$

$$\text{EQ:7} \quad [R + Q]^\eta = 1$$

$$\text{EQ:8} \quad R^\eta + \eta R^{\eta-1} \cdot Q + \frac{\eta(\eta-1)R^{\eta-2} \cdot Q^2}{2!} + \frac{\eta(\eta-1)(\eta-2)R^{\eta-3} \cdot Q^3}{3!} + \dots + Q^\eta = 1$$

$$\text{EQ:9} \quad R_T(t) = \underbrace{R}^{\text{1st}} + \underbrace{\eta R^{\eta-1} \cdot Q}_{\text{2nd}} - \text{binomial terms}$$

$$\text{EQ:10} \quad R(t)_{\text{system}} = [R(t)_{\text{page}}]^P$$

$$\text{EQ:11} \quad R(t)_{\text{PAGE}_{\text{nec}}} = R(t)_{\text{DEV}_{\text{nec}}}^\eta = e^{-\lambda \cdot \eta \cdot t}$$

$$\text{EQ:12} \quad R(t)_{\text{SYS}_{\text{nec}}} = [R(t)_{\text{PAGE}_{\text{nec}}}]^P = [R(t)_{\text{DEV}}]^{P \cdot \eta}$$

$$\text{EQ:13} \quad R(t)_{\text{PAGE}_{\text{ecc}}} = [R(t)_x^\eta + \eta R(t)_x^{\eta-1} \cdot Q(t)_x]^{\ell_x}$$

$$\text{EQ:14a} \quad RT_1 = R_{f_1} + \eta R_{f_1}^{\eta-1} \cdot Q_{f_1}$$

$$\text{EQ:14b} \quad RT_2 = R_{f_2} \cdot [RT_1]^{\ell_2} + \eta [R_{f_2} \cdot R_{f_1}^{\ell_2}]^{\eta-1} \cdot Q_{f_2}$$

$$\text{EQ:14c} \quad \lambda_{f_1} = \lambda_{\text{dev}} \cdot X_{f_1} \cdot \frac{\text{fsz}_{f_1}}{\text{MSZ}} \quad \lambda_{f_2} = \lambda_{\text{dev}} \cdot X_{f_2} \cdot \frac{\text{fsz}_{f_2}}{\text{MSZ}}$$

$$\text{EQ:14d} \quad R(t)_{\text{page}} = [RT_2]^{\frac{\text{MSZ}}{\text{fsz}_2}}$$

$$\text{EQ:15} \quad RT_3 = R_{f_3} \cdot [RT_2]^{\ell_3} + \eta [R_{f_3} (R_{f_2} (R_{f_1}^{\ell_2})^{\ell_3})]^{\eta-1} \cdot Q_{f_3}$$

$$\text{EQ:16} \quad \left[i \leftarrow \begin{array}{l} \ell_i = \frac{\text{fsz}_i}{\text{fsz}_{i-1}} \\ \lambda_{f_i} = \lambda_{\text{dev}} \cdot X_i \cdot \frac{\text{fsz}_i}{\text{MSZ}} \end{array} \right]$$

$$R(t)_{\text{PAGE}_{\text{ecc}}} = \left\{ i \leftarrow \begin{array}{l} R_i = e^{-\lambda_{f_i} \cdot t}, Q_i = 1 - R_i \\ RS_i = R_i \cdot (RS_{i-1})^{\ell_i} \\ RT_i = R_i^\eta \cdot (RT_{i-1})^{\ell_i} + \eta (RS_i)^{\eta-1} \cdot Q_i \end{array} \right\} \left\{ \begin{array}{l} \text{MSZ} \\ \text{fsz}_N \end{array} \right\}$$

$$R(t)_{\text{SYSTEM}_{\text{ecc}}} = [R(t)_{\text{PAGE}_{\text{ecc}}}]^{\text{Pages}}$$

$$\text{restrictions: } RS_0 = RT_0 = \text{fsz}_0 = 1.$$

$$\text{EQ:17} \quad \text{MTBF}_{\text{sys}} = \int_0^{\infty} R(t)_{\text{sys}} \cdot dt$$

$$\text{EQ:18} \quad \text{EF} = \frac{\text{MTBF}_{\text{sys-ecc}}}{\text{MTBF}_{\text{sys-necc}}}$$

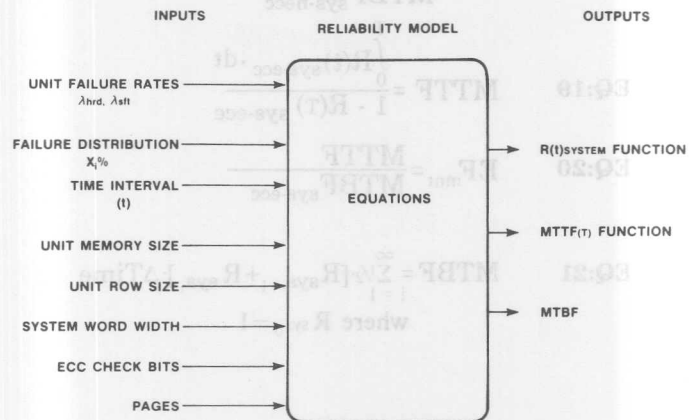
$$\text{EQ:19} \quad \text{MTTF} = \frac{\int_0^T R(t)_{\text{sys-ecc}} \cdot dt}{1 - R(T)_{\text{sys-ecc}}}$$

$$\text{EQ:20} \quad \text{EF}_{\text{mnt}} = \frac{\text{MTTF}}{\text{MTBF}_{\text{sys-ecc}}}$$

$$\text{EQ:21} \quad \text{MTBF} = \sum_{i=1}^{\infty} \frac{1}{2} [R_{\text{sys}_{i-1}} + R_{\text{sys}_i}] \cdot \Delta \text{Time}$$

where $R_{\text{sys}_0} = 1$

APPENDIX B



SYS /01

ECC PROBABILITY PROGRAM "INTEL-MPD/MC."

MEMORY SYS: SIZE-> 32.0KB WORD WIDTH-> 16 + 6 NO. PAGES-> 1 X 1
 COMPONENT: TOTAL-> 16 + 6 RAM SIZE-> 16384 COL SIZE-> 128
 SYSTEM DATA: TTL RATE -> 0.00000%/1K-HRS, SYSTEM RATE -> 0.00000%/1K-HRS

FAILURE DATA: MTBF NECC -> 49212.60HRS, MTBF SYS -> 0.00HRS
 HARD ERRORS: PARTIAL -> 0 CELLS/PG RATE -> 0.027000% / 1000 HRS
 SOFT ERRORS: "*" MAINT -> 0 HRS, RATE -> 0.100000% / 1000 HRS
 ANALYSIS DATA: PERIOD -> 100000.00HRS, AVE CELL FAILURE -> 16.2

FAILURE TYPE RATIOS:
 -TYPE- =DISTRIBUTION=GEOMETRY=UNIT. RATE/1K HRS=AVE. CELLS=ECC. DISTR=EXPS=
 SOFT ERROR -> [78.740%] / 16384. = 0.61035E-05%, 0.787 [4.87%]
 HARD ERRORS -> [21.260%]
 SINGLE CELL -> 50.0000% / 16384. = 0.82397E-06%, 0.106 0.69% 1.
 ROW OR COL -> 43.7000% / 128 = 0.92180E-04%, 11.892 77.36% 128.
 COLUMN/ROW -> 6.2000% / 64 = 0.26156E-04%, 3.374 21.95% 2.
 HALF CHIP -> 0.0000% / 2 = 0.00000E+00%, 0.000 0.00% 32.
 TOTAL CHIP -> 0.0000% / 1 = 0.00000E+00%, 0.000 0.00% 2.

PERIOD	PMET	RITJ SECC	MTTF	ENHANCEMENT	% - R(T)
<- CM -	<- HRS>	=FUNCTION=	< HRS>	FACTOR	@T
0	0.00	1.00000	0	0	100.0%
1	0.10	0.99332	14922026	303	99.3%
2	0.20	0.97389	7584078	154	97.4%
3	0.30	0.94293	5149121	105	94.3%
4	0.40	0.90200	3939900	80	90.2%
5	0.50	0.85289	3221005	65	85.3%
6	0.60	0.79748	2747323	56	79.7%
7	0.70	0.73770	2413825	49	73.8%
8	0.80	0.67537	2168007	44	67.5%
9	0.90	0.61217	1980705	40	61.2%
10	1.00	0.54958	1834422	37	55.0%
11	1.10	0.48884	1718017	35	48.9%
12	1.20	0.43095	1624061	33	43.1%
13	1.30	0.37666	1547397	31	37.7%
14	1.40	0.32650	1484334	30	32.6%
15	1.50	0.28075	1432149	29	28.1%
16	1.60	0.23956	1388787	28	24.0%
17	1.70	0.20290	1352665	27	20.3%
18	1.80	0.17062	1322533	27	17.1%
19	1.90	0.14248	1297395	26	14.2%
20	2.00	0.11819	1276437	26	11.8%
21	2.10	0.09741	1258993	26	9.7%
22	2.20	0.07979	1244505	25	8.0%
23	2.30	0.06496	1232508	25	6.5%
24	2.40	0.05258	1222606	25	5.3%
25	2.50	0.04232	1214463	25	4.2%
26	2.60	0.03388	1207796	25	3.4%
27	2.70	0.02698	1202358	24	2.7%
28	2.80	0.02137	1197945	24	2.1%
29	2.90	0.01685	1194379	24	1.7%
30	3.00	0.01322	1191512	24	1.3%

 31 3.00 0.01322 =MEMORY MTBF= =EF= 1175755.35 24.
 FIN 1 =SYSTEM MTBF= 1175755.35 24. 0.85052E-03

SYS /01

ECC PROBABILITY PROGRAM "INTEL-MPD/MC."

MEMORY SYS: SIZE-> 4.1MB WORD WIDTH-> 16 + 6 NO. PAGES-> 1 X128
 COMPONENT: TOTAL-> 2048 + 768 RAM SIZE-> 16384 COL SIZE-> 128
 SYSTEM DATA: TTL RATE -> 0.00000%/1K-HRS, SYSTEM RATE -> 0.00000%/1K-HRS

FAILURE DATA: MTBF NECC -> 384.47HRS, MTBF SYS -> 0.00HRS
 HARD ERRORS: PARTIAL -> 0 CELLS/PG RATE -> 0.027000% / 1000 HRS
 SOFT ERRORS: "*" MAINT -> 0 HRS, RATE -> 0.100000% / 1000 HRS
 ANALYSIS DATA: PERIOD -> 8000.00HRS, AVE CELL FAILURE -> 16.2

FAILURE TYPE RATIOS:
 -TYPE- =DISTRIBUTION=GEOMETRY=UNIT. RATE/1K HRS=AVE. CELLS=ECC. DISTR=EXPS=
 SOFT ERROR -> [78.740%] / 16384. = 0.61035E-05%, 0.787 [4.87%]
 HARD ERRORS -> [21.260%]
 SINGLE CELL -> 50.0000% / 16384. = 0.82397E-06%, 0.106 0.69% 1.
 ROW OR COL -> 43.7000% / 128 = 0.92180E-04%, 11.892 77.36% 128.
 COLUMN/ROW -> 6.2000% / 64 = 0.26156E-04%, 3.374 21.95% 2.
 HALF CHIP -> 0.0000% / 2 = 0.00000E+00%, 0.000 0.00% 32.
 TOTAL CHIP -> 0.0000% / 1 = 0.00000E+00%, 0.000 0.00% 2.

PERIOD	PMET	RITJ SECC	MTTF	ENHANCEMENT	% - R(T)
<- CM -	<- HRS>	=FUNCTION=	< HRS>	FACTOR	@T
0	0.00	1.00000	0	0	100.0%
1	8000	0.99446	1439611	3744	99.4%
2	16000	0.97804	722580	1879	97.8%
3	24000	0.95132	484470	1260	95.1%
4	32000	0.91518	366101	952	91.5%
5	40000	0.87080	295638	769	87.1%
6	48000	0.81955	249140	648	82.0%
7	56000	0.76294	216349	563	76.3%
8	64000	0.70256	192135	500	70.3%
9	72000	0.63998	173652	452	64.0%
10	80000	0.57670	159190	414	57.7%
11	88000	0.51411	147663	384	51.4%
12	96000	0.45341	138346	360	45.3%
13	104000	0.39562	130737	340	39.6%
14	112000	0.34153	124475	324	34.2%
15	120000	0.29171	119297	310	29.2%
16	128000	0.24653	115001	299	24.7%
17	136000	0.20616	111433	290	20.6%
18	144000	0.17059	108471	282	17.1%
19	152000	0.13968	106017	276	14.0%
20	160000	0.11318	103990	270	11.3%
21	168000	0.09076	102322	266	9.1%
22	176000	0.07202	100958	263	7.2%
23	184000	0.05656	99849	260	5.7%
24	192000	0.04397	98954	257	4.4%
25	200000	0.03382	98237	256	3.4%
26	208000	0.02575	97668	254	2.6%
27	216000	0.01941	97220	253	1.9%
28	224000	0.01448	96872	252	1.4%
29	232000	0.01069	96603	251	1.1%
30	240000	0.00782	96397	251	0.8%

 31 240000.00 0.00782 =MEMORY MTBF= =EF= 95643.55 249
 FIN 1 =SYSTEM MTBF= 95643.55 249 0.10455E-01

SYS /Q1

ECC PROBABILITY PROGRAM "INTEL-MPD/MC."

MEMORY SYS: SIZE-> 64.0KB WORD WIDTH-> 32. + 7. NO. PAGES-> 1. X 1.
 COMPONENT: TOTAL-> 32 + 7 RAM SIZE-> 16384. COL SIZE-> 128.
 SYSTEM DATA: TTL RATE -> 0.00000%/1K-HRS, SYSTEM RATE -> 0.00000%/1K-HRS

FAILURE DATA: MTBF, NECC -> 24606.30HRS, MTBF, SYS -> 0.00HRS
 HARD ERRORS: PARTIAL -> 0. CELLS/PG RATE -> 0.027000% / 1000 HRS
 SOFT ERRORS: "*" MAINT -> 0. HRS, RATE -> 0.100000% / 1000 HRS
 ANALYSIS DATA: PERIOD -> 66000.00HRS, AVE CELL FAILURE -> 16.2

FAILURE TYPE RATIOS:

=TYPE= =DISTRIBUTION=GEOMETRY=UNIT. RATE/1K HRS=AVE. CELLS=ECC. DISTR=EXPS=

SOFT ERROR -> [78.740%] / 16384. = 0.61035E-05%, 0.787 [4.87%]
 HARD ERRORS -> [21.260%] [95.13%]
 SINGLE CELL -> 50.0000% / 16384. = 0.82397E-06%, 0.106 0.69% 1.
 ROW OR COL -> 43.7000% / 128. = 0.92180E-04%, 11.892 77.36% 128.
 COLUMN/ROW -> 6.2000% / 64. = 0.26156E-04%, 3.374 21.95% 2.
 HALF CHIP -> 0.0000% / 2. = 0.00000E+00%, 0.000 0.00% 32.
 TOTAL CHIP -> 0.0000% / 1. = 0.00000E+00%, 0.000 0.00% 2.

PERIOD	PMET:	R(T). SECC	MTTF	ENHANCEMENT	% - R(T)
-----	< - HRS>	=FUNCTION=	< HRS>	FACTOR	@
0	0.	1.00000	0.	0	100.0%
1	66000.	0.99070	7066011.	287.	99.1%
2	132000.	0.96388	3604737.	146.	96.4%
3	198000.	0.92173	2458257.	100.	92.2%
4	264000.	0.86699	1890484.	77.	86.7%
5	330000.	0.80276	1554231.	63.	80.3%
6	396000.	0.73219	1333790.	54.	73.2%
7	462000.	0.65828	1179584.	48.	65.8%
8	528000.	0.58374	1066829.	43.	58.4%
9	594000.	0.51088	981755.	40.	51.1%
10	660000.	0.44151	916096.	37.	44.2%
11	726000.	0.37700	864581.	35.	37.7%
12	792000.	0.31821	823686.	33.	< 31.8%>
13	858000.	0.26564	790958.	32.	26.6%
14	924000.	0.21942	764629.	31.	21.9%
15	990000.	0.17941	743388.	30.	17.9%
16	1056000.	0.14528	726238.	30.	14.5%
17	1122000.	0.11655	712400.	29.	11.7%
18	1188000.	0.09267	701259.	28.	9.3%
19	1254000.	0.07305	692319.	28.	7.3%
20	1320000.	0.05712	685174.	28.	5.7%
21	1386000.	0.04431	679492.	28.	4.4%
22	1452000.	0.03411	674998.	27.	3.4%
23	1518000.	0.02607	671465.	27.	2.6%
24	1584000.	0.01979	668705.	27.	2.0%
25	1650000.	0.01492	666562.	27.	1.5%
26	1716000.	0.01118	664910.	27.	1.1%
27	1782000.	0.00832	663644.	27.	0.8%
			=MEMORY MTBF=	=EF=	
28	1782000.00	0.00832	658122.51	27.	

FIN 1 =SYSTEM MTBF= 658122.51 27. 0.15195E-02

SYS /Q1

ECC PROBABILITY PROGRAM "INTEL-MPD/MC."

MEMORY SYS: SIZE-> 8.2MB WORD WIDTH-> 32. + 7. NO. PAGES-> 1. X128.
 COMPONENT: TOTAL-> 4096 + 896. RAM SIZE-> 16384. COL SIZE-> 128.
 SYSTEM DATA: TTL RATE -> 0.00000%/1K-HRS, SYSTEM RATE -> 0.00000%/1K-HRS

FAILURE DATA: MTBF, NECC -> 192.24HRS, MTBF, SYS -> 0.00HRS
 HARD ERRORS: PARTIAL -> 0. CELLS/PG RATE -> 0.027000% / 1000 HRS
 SOFT ERRORS: "*" MAINT -> 0. HRS, RATE -> 0.100000% / 1000 HRS
 ANALYSIS DATA: PERIOD -> 5000.00HRS, AVE CELL FAILURE -> 16.2

FAILURE TYPE RATIOS:

=TYPE= =DISTRIBUTION=GEOMETRY=UNIT. RATE/1K HRS=AVE. CELLS=ECC. DISTR=EXPS=

SOFT ERROR -> [78.740%] / 16384. = 0.61035E-05%, 0.787 [4.87%]
 HARD ERRORS -> [21.260%] [95.13%]
 SINGLE CELL -> 50.0000% / 16384. = 0.82397E-06%, 0.106 0.69% 1.
 ROW OR COL -> 43.7000% / 128. = 0.92180E-04%, 11.892 77.36% 128.
 COLUMN/ROW -> 6.2000% / 64. = 0.26156E-04%, 3.374 21.95% 2.
 HALF CHIP -> 0.0000% / 2. = 0.00000E+00%, 0.000 0.00% 32.
 TOTAL CHIP -> 0.0000% / 1. = 0.00000E+00%, 0.000 0.00% 2.

PERIOD	PMET:	R(T). SECC	MTTF	ENHANCEMENT	% - R(T)
-----	< - HRS>	=FUNCTION=	< HRS>	FACTOR	@
0	0.	1.00000	0.	0	100.0%
1	5000.	0.99306	718156.	3736.	99.3%
2	10000.	0.97257	360766.	1877.	97.3%
3	15000.	0.93940	242201.	1260.	93.9%
4	20000.	0.89494	183350.	954.	89.5%
5	25000.	0.84095	148393.	772.	84.1%
6	30000.	0.77946	125392.	652.	77.9%
7	35000.	0.71269	109232.	568.	71.3%
8	40000.	0.64283	97356.	506.	64.3%
9	45000.	0.57202	88344.	460.	57.2%
10	50000.	0.50219	81346.	423.	50.2%
11	55000.	0.43499	75818.	394.	43.5%
12	60000.	0.37176	71398.	371.	37.2%
13	65000.	0.31351	67835.	353.	< 31.4%>
14	70000.	0.26090	64949.	338.	26.1%
15	75000.	0.21425	62605.	326.	21.4%
16	80000.	0.17364	60702.	316.	17.4%
17	85000.	0.13888	59159.	308.	13.9%
18	90000.	0.10963	57913.	301.	11.0%
19	95000.	0.08542	56913.	296.	8.5%
20	100000.	0.06569	56116.	292.	6.6%
21	105000.	0.04987	55486.	289.	5.0%
22	110000.	0.03737	54992.	286.	3.7%
23	115000.	0.02765	54609.	284.	2.8%
24	120000.	0.02019	54316.	283.	2.0%
25	125000.	0.01456	54093.	281.	1.5%
26	130000.	0.01037	53927.	281.	1.0%
27	135000.	0.00729	53804.	280.	0.7%
			=MEMORY MTBF=	=EF=	
28	135000.00	0.00729	53412.22	278.	

FIN 1 =SYSTEM MTBF= 53412.22 278. 0.18722E-01

SYS /Q1 ECC PROBABILITY PROGRAM "INTEL-MPD/MC."

MEMORY SYS: SIZE-> 128.0KB WORD WIDTH-> 64. + 8. NO. PAGES-> 1. X 1.
 COMPONENT: TOTAL-> 64. + 8. RAM SIZE-> 16384. COL SIZE-> 128.
 SYSTEM DATA: TTL RATE -> 0.00000%/1K-HRS, SYSTEM RATE -> 0.00000%/1K-HRS

FAILURE DATA: MTBF, NECC -> 12303.15HRS, MTBF, SYS -> 0.00HRS
 HARD ERRORS: PARTIAL -> 0. CELLS/PG RATE -> 0.027000% / 1000 HRS
 SOFT ERRORS: "*" MAINT -> 0. HRS, RATE -> 0.100000% / 1000 HRS
 ANALYSIS DATA: PERIOD -> 33000.00HRS, AVE CELL FAILURE -> 16.2

FAILURE TYPE RATIOS:-----
 =TYPE= =DISTRIBUTION=GEOMETRY=UNIT. RATE/1K HRS=AVE. CELLS=ECC. DISTR=EXPS=
 SOFT ERROR -> [78.740%] / 16384. = 0.61035E-05%, 0.787 [4.87%]
 HARD ERRORS -> [21.260%] [95.13%]
 SINGLE CELL -> 50.0000% / 16384. = 0.82397E-06%, 0.106 0.69% 1.
 ROW OR COL -> 43.7000% / 128. = 0.92180E-04%, 11.892 77.36% 128.
 COLUMN/ROW -> 6.2000% / 64. = 0.26156E-04%, 3.374 21.95% 2.
 HALF CHIP -> 0.0000% / 2. = 0.00000E+00%, 0.000 0.00% 32.
 TOTAL CHIP -> 0.0000% / 1. = 0.00000E+00%, 0.000 0.00% 2.

PERIOD	PMET	RTI, SECC	MTTF	ENHANCEMENT	% - R(T)
-----	< - HRS>	=FUNCTION=	< HRS >	FACTOR	ET
0	0.	1.00000	0.	0.	100.0%
1	33000.	0.99197	4092667.	333.	99.2%
2	66000.	0.96871	2084463.	169.	96.9%
3	99000.	0.93192	1418701.	115.	93.2%
4	132000.	0.88376	1088549.	88.	88.4%
5	165000.	0.82663	892655.	73.	82.7%
6	198000.	0.76308	763911.	62.	76.3%
7	231000.	0.69556	673564.	55.	69.6%
8	264000.	0.62639	607238.	49.	62.6%
9	297000.	0.55758	556949.	45.	55.8%
10	330000.	0.49083	517906.	42.	49.1%
11	363000.	0.42747	487054.	40.	42.7%
12	396000.	0.36848	462357.	38.	< 36.8%>
13	429000.	0.31451	442398.	36.	31.5%
14	462000.	0.26592	426159.	35.	26.6%
15	495000.	0.22280	412889.	34.	22.3%
16	528000.	0.18504	402018.	33.	18.5%
17	561000.	0.15240	393104.	32.	15.2%
18	594000.	0.12450	385797.	31.	12.5%
19	627000.	0.10093	379818.	31.	10.1%
20	660000.	0.08120	374936.	30.	8.1%
21	693000.	0.06487	370964.	30.	6.5%
22	726000.	0.05146	367744.	30.	5.1%
23	759000.	0.04056	365147.	30.	4.1%
24	792000.	0.03176	363061.	30.	3.2%
25	825000.	0.02472	361395.	29.	2.5%
26	858000.	0.01912	360071.	29.	1.9%
27	891000.	0.01471	359025.	29.	1.5%
28	924000.	0.01125	358203.	29.	1.1%
29	957000.	0.00856	357561.	29.	0.9%
30	957000.00	0.00856	=MEMORY MTBF=	=EF=	
			354499.33	29.	

FIN 1 =SYSTEM MTBF= 354499.33 29 0.28209E-02

xxiii

SYS /Q1 ECC PROBABILITY PROGRAM "INTEL-MPD/MC."

MEMORY SYS: SIZE-> 16.4MB WORD WIDTH-> 64. + 8. NO. PAGES-> 1. X128.
 COMPONENT: TOTAL-> 8192. + 1024. RAM SIZE-> 16384. COL SIZE-> 128.
 SYSTEM DATA: TTL RATE -> 0.00000%/1K-HRS, SYSTEM RATE -> 0.00000%/1K-HRS

FAILURE DATA: MTBF, NECC -> 96.12HRS, MTBF, SYS -> 0.00HRS
 HARD ERRORS: PARTIAL -> 0. CELLS/PG RATE -> 0.027000% / 1000 HRS
 SOFT ERRORS: "*" MAINT -> 0. HRS, RATE -> 0.100000% / 1000 HRS
 ANALYSIS DATA: PERIOD -> 2500.00HRS, AVE CELL FAILURE -> 16.2

FAILURE TYPE RATIOS:-----
 =TYPE= =DISTRIBUTION=GEOMETRY=UNIT. RATE/1K HRS=AVE. CELLS=ECC. DISTR=EXPS=
 SOFT ERROR -> [78.740%] / 16384. = 0.61035E-05%, 0.787 [4.87%]
 HARD ERRORS -> [21.260%] [95.13%]
 SINGLE CELL -> 50.0000% / 16384. = 0.82397E-06%, 0.106 0.69% 1.
 ROW OR COL -> 43.7000% / 128. = 0.92180E-04%, 11.892 77.36% 128.
 COLUMN/ROW -> 6.2000% / 64. = 0.26156E-04%, 3.374 21.95% 2.
 HALF CHIP -> 0.0000% / 2. = 0.00000E+00%, 0.000 0.00% 32.
 TOTAL CHIP -> 0.0000% / 1. = 0.00000E+00%, 0.000 0.00% 2.

PERIOD	PMET	RTI, SECC	MTTF	ENHANCEMENT	% - R(T)
-----	< - HRS>	=FUNCTION=	< HRS >	FACTOR	ET
0	0. M-	1.00000	0.	0.	100.0%
1	2500.	0.99401	416361.	4332.	99.4%
2	5000.	0.97629	209042.	2175.	97.6%
3	7500.	0.94751	140217.	1459.	94.8%
4	10000.	0.90869	106020.	1103.	90.9%
5	12500.	0.86119	85676.	891.	86.1%
6	15000.	0.80658	72264.	752.	80.7%
7	17500.	0.74658	62816.	654.	74.7%
8	20000.	0.68298	55851.	581.	68.3%
9	22500.	0.61752	50543.	526.	61.8%
10	25000.	0.55187	46400.	483.	55.2%
11	27500.	0.48749	43106.	448.	48.7%
12	30000.	0.42566	40453.	421.	42.6%
13	32500.	0.36741	38295.	398.	< 36.7%>
14	35000.	0.31350	36528.	380.	31.3%
15	37500.	0.26445	35074.	365.	26.4%
16	40000.	0.22053	33876.	352.	22.1%
17	42500.	0.18183	32888.	342.	18.2%
18	45000.	0.14822	32075.	334.	14.8%
19	47500.	0.11947	31407.	327.	11.9%
20	50000.	0.09521	30862.	321.	9.5%
21	52500.	0.07503	30419.	316.	7.5%
22	55000.	0.05847	30061.	313.	5.8%
23	57500.	0.04506	29774.	310.	4.5%
24	60000.	0.03434	29546.	307.	3.4%
25	62500.	0.02588	29367.	306.	2.6%
26	65000.	0.01929	29227.	304.	1.9%
27	67500.	0.01422	29120.	303.	1.4%
28	70000.	0.01037	29037.	302.	1.0%
29	72500.	0.00748	28975.	301.	0.7%
30	72500.00	0.00748	=MEMORY MTBF=	=EF=	
			28758.48	299.	

FIN 1 =SYSTEM MTBF= 28758.48 299. 0.34772E-01

xxiv

INTRODUCTION

HMOS II is Intel's highest performance N-channel MOS technology. As in the previous generation HMOS, high performance has been achieved by device scaling techniques. Intel's new generation of high speed static RAMs (2115H/25H, 2147H, 2148H/49H), have been designed on this process.

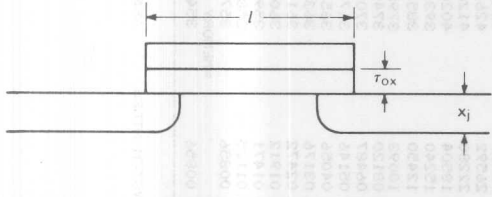
Data presented in this report show this advanced technology yields products as reliable as HMOS; reliability testing has shown no increase in device failure rate due to device scaling. A failure rate of 0.014%/1000 hours at 70°C (60% UCL) has been calculated for HMOS II products as compared to 0.020%/1000 hours.

This reliability report is organized into 5 sections: Technology Description, Failure Mechanisms, Reliability Testing, Test Results, and Summary.

TECHNOLOGY DESCRIPTION

The high performance of HMOS II is achieved by combining MOS device scaling with on-chip substrate bias generation. By reducing the physical parameters by a fixed scaling factor, circuit density and performance were increased while decreasing active circuit power. As shown in Table 1, HMOS II uses polysilicon gate lengths down to 2.0μ and a gate oxide thickness of 400\AA . Shallow junctions ($<0.8\mu\text{m}$) are obtained by using arsenic source-drain diffusant. In addition, oxide isolation and depletion load processing are employed to improve circuit performance and density. The technology figure of merit, the speed-power product, was measured to be 0.5 pJ using an 11 stage ring oscillator with speed fanout of 3.^(1,2)

Table 1. MOS Technology Evolution

			
Parameter	MOS, 1976	HMOS, 1977	HMOS II, 1979
Channel Length, $l(\mu\text{m})$	6	3.5	2.0
Gate Oxide Thickness, $\tau_{ox}(\text{\AA})$	1,100	700	400
Junction Depth, $x_j(\mu)$	1.7	<1.0	<0.8
Depletion Loads	No	Yes	Yes
Oxide Isolation	No	Yes	Yes
Built-in Substrate Bias	Yes	Yes	Yes
Speed Power Product (pJ)	4.0	1.0	0.5

Intel's family of HMOS II products is shown in Table 2. Each of the HMOS II static RAMs listed is designed with the same design rules and utilizes the same 6-transistor static memory cell and substrate bias generator. Much of the other internal circuitry is similar. For these reasons, the reliability of all HMOS II devices will exhibit similar characteristics.

Table 2. HMOS II RAMs

Device	Density	Configuration	Access Time (ns) (max.)
2115H/25H	1K	$1K \times 1$	25-35
2147H	4K	$4K \times 1$	35-55
2148H/49H	4K	$1K \times 4$	45-70

The 6-transistor static memory cell used as the storage medium is shown in Figure 1. This cross-coupled flip-flop uses depletion load pullups to increase speed and is the same basic memory cell used in previous generation static RAMs. Figures 2, 3, and 4 show pinout description, block diagram, and address map for the 2115H/25H, 2147H and the 2148H/49H, respectively.

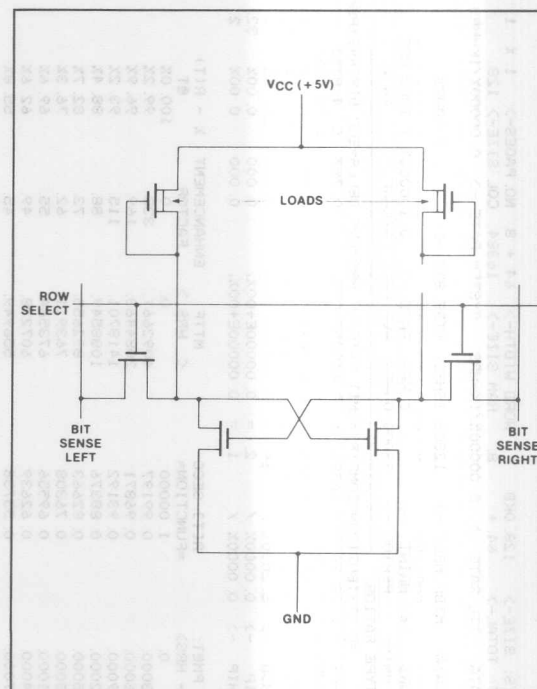


Figure 1. Cell Schematic for HMOS II Static RAMs

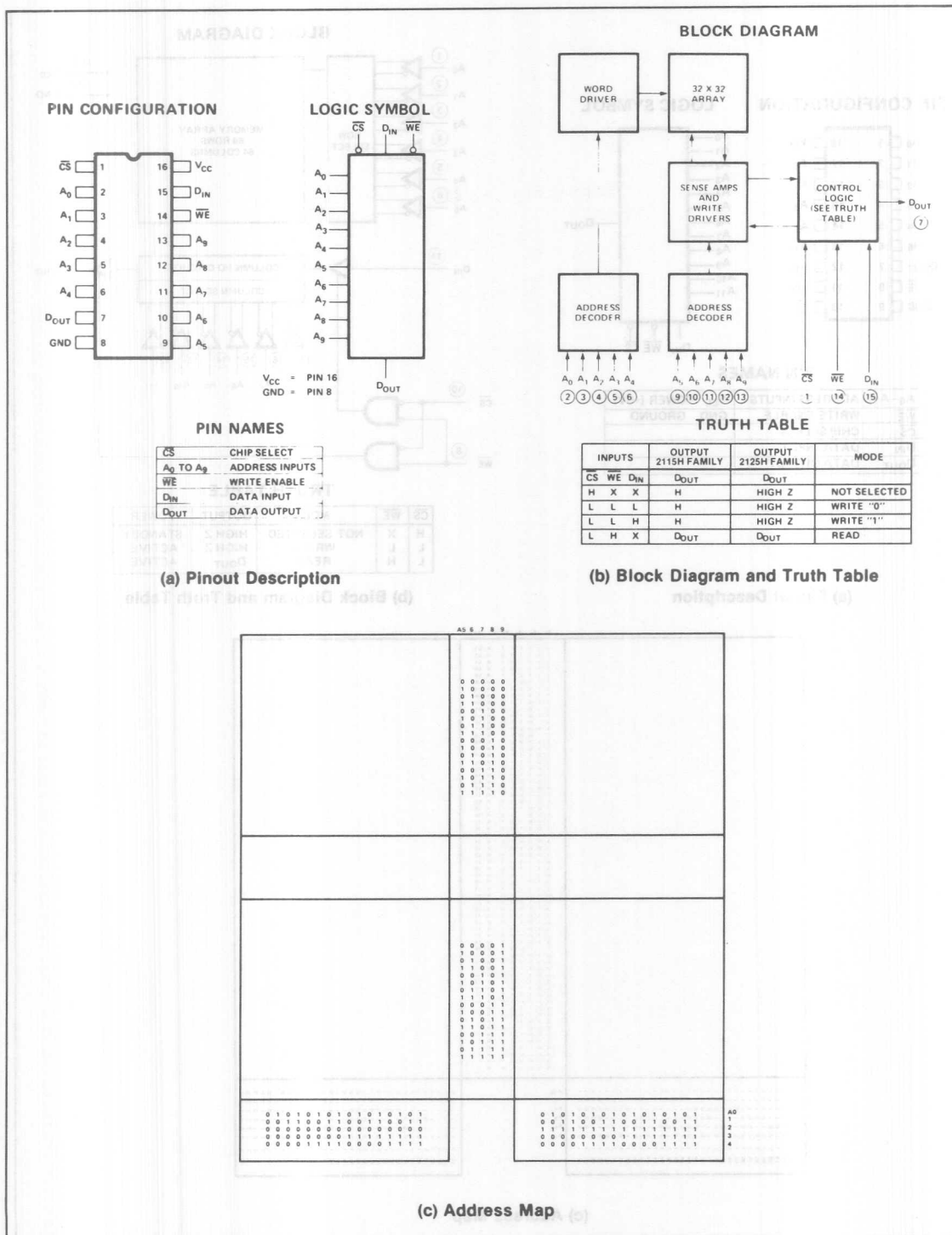


Figure 2. 2115/25H Device Description

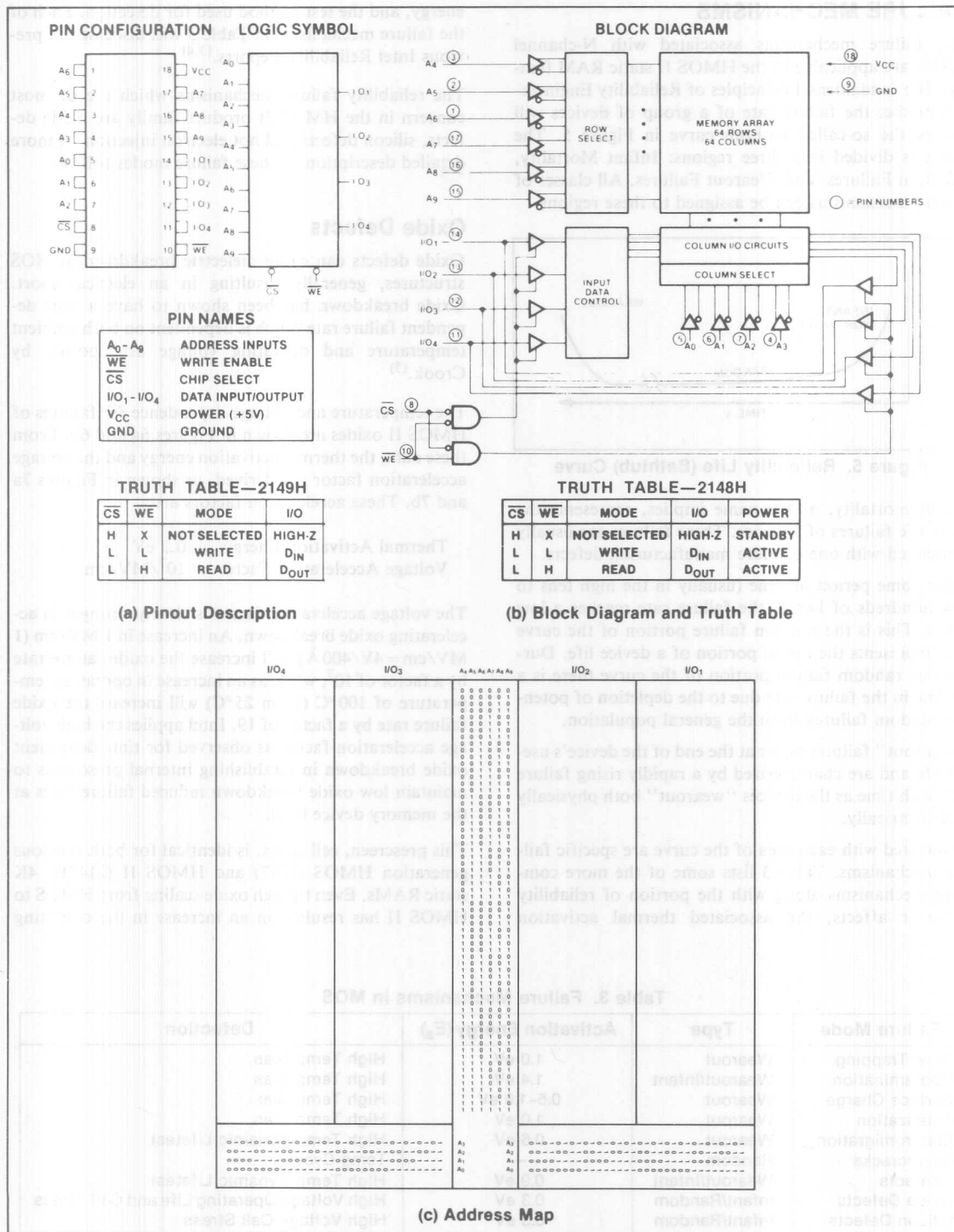


Figure 4. 2148H/49H Device Description

FAILURE MECHANISMS

The failure mechanisms associated with N-channel RAMs are applicable to the HMOS II static RAM family. The fundamental principles of Reliability Engineering predict the failure rate of a group of devices will follow the so-called bathtub curve in Figure 5. The curve is divided into three regions: Infant Mortality, Random Failures, and Wearout Failures. All classes of failure mechanisms can be assigned to these regions.

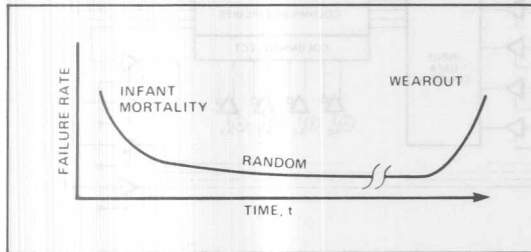


Figure 5. Reliability Life (Bathtub) Curve

Infant mortality, as the name implies, represents the early life failures of a device. These failures are usually associated with one or more manufacturing defects.

After some period of time (usually in the high tens to low hundreds of hours) the failure rate reaches a low value. This is the random failure portion of the curve that represents the useful portion of a device life. During this random failure portion of the curve there is a decline in the failure rate due to the depletion of potential random failures from the general population.

“Wearout” failures occur at the end of the device’s useful life and are characterized by a rapidly rising failure rate with time as the devices “wearout” both physically and electrically.

Associated with each area of the curve are specific failure mechanisms. Table 3 lists some of the more common mechanisms along with the portion of reliability curve it affects, the associated thermal activation

energy, and the test method used for detection. Each of the failure mechanisms in Table 3 was discussed in previous Intel Reliability Reports.^(3,4)

The reliability failure mechanisms which are of most concern in the HMOS II product family are oxide defects, silicon defects and hot electron injection. A more detailed description of these failure modes follows.

Oxide Defects

Oxide defects can cause dielectric breakdown in MOS structures, generally resulting in an electrical short. Oxide breakdown has been shown to have a time dependent failure rate which is dependent on both ambient temperature and operating voltage as reported by Crook.⁽⁵⁾

The temperature and voltage dependence for failures of HMOS II oxides are shown in Figures 6a and 6b. From these data, the thermal activation energy and the voltage acceleration factor are derived, as shown in Figures 7a and 7b. These acceleration factors are:

Thermal Activation Energy	0.3 eV
Voltage Acceleration Factor	$10^7/\text{MV/cm}$

The voltage acceleration factor is clearly stronger in accelerating oxide breakdown. An increase in 1 MV/cm (1 MV/cm = 4V/400 Å) will increase the oxide failure rate by a factor of 10^7 , whereas an increase in operating temperature of 100°C (from 25°C) will increase the oxide failure rate by a factor of 19. Intel applies the high voltage acceleration factor as observed for time dependent oxide breakdown in establishing internal prescreens to maintain low oxide breakdown induced failure rates at the memory device level.

This prescreen, cell stress, is identical for both previous generation HMOS (2147) and HMOS II (2147H) 4K static RAMs. Even though oxide scaling from HMOS to HMOS II has resulted in an increase in the operating

Table 3. Failure Mechanisms in MOS

Failure Mode	Type	Activation Energy (E_a)	Detection
Slow Trapping	Wearout	1.0 eV	High Temp Bias
Contamination	Wearout/Infant	1.4 eV	High Temp Bias
Surface Charge	Wearout	0.5–1.0 eV	High Temp Bias
Polarization	Wearout	1.0 eV	High Temp Bias
Electromigration	Wearout	0.6 eV	High Temp Dynamic Lifetest
Microcracks	Random	—	Temp Cycling
Contacts	Wearout/Infant	0.9 eV	High Temp Dynamic Lifetest
Oxide Defects	Infant/Random	0.3 eV	High Voltage Operating Life and Cell Stress
Silicon Defects	Infant/Random	0.3 eV	High Voltage Cell Stress
Electron Injection	Wearout	—	Low Temp High Voltage Operating Life

fields across the gate oxide from 0.71 MV/cm to 1.25 MV/cm, the resulting device failure rates after cell stress are equivalent.

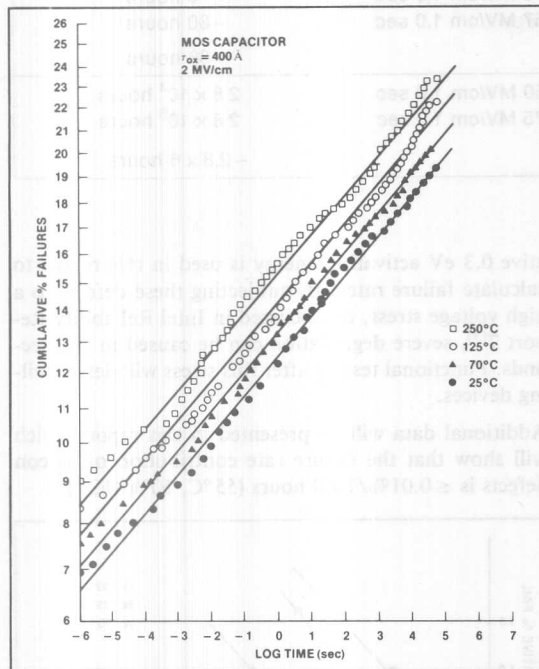


Figure 6a. Large Area MOS Capacitor Time Dependent Dielectric Breakdown Data for Different Ambient Temperatures

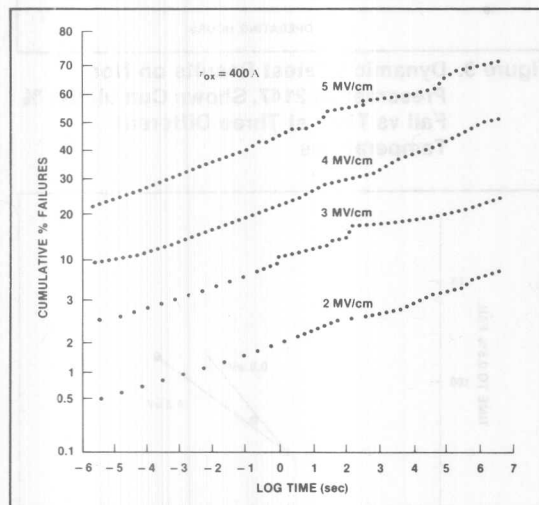


Figure 6b. Large Area MOS Capacitor Time Dependent Oxide Breakdown Data for HMOS II Oxides at Various Stress Fields

This is achieved as a result of the increased stress field during device cell stress which gives a significantly longer equivalent aging time for HMOS II oxides. Table 4 demonstrates the effect of cell stress on device aging.

Figure 8 graphically illustrates how cell stress aging decreases device failure rates. This cell stress prescreen is performed on 100% of the product during electrical testing in the standard manufacturing flow.

HMOS II static RAM oxide failure rates can be derived from Intel's standard 125°C lifestest data. These data indicate that the device oxide failure rate at 55°C for HMOS II is <0.01%/K hours.

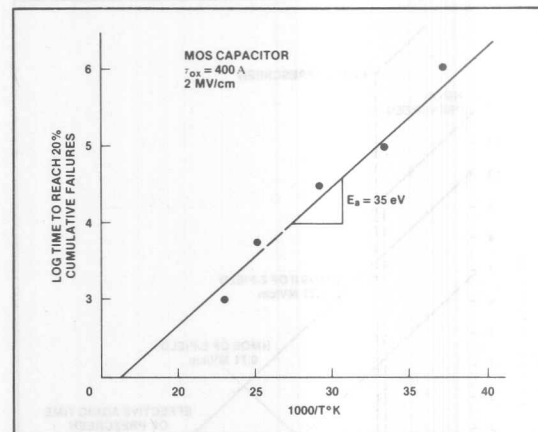


Figure 7a. Arrhenius Plot for Time Dependent Dielectric Breakdown

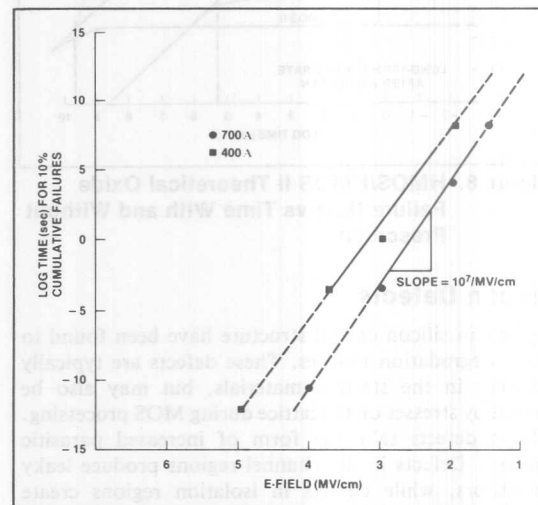


Figure 7b. Log Time to Reach 10% Cumulative Failures as a Function of Electric Field for HMOS (700 Å) and HMOS II (400 Å) Oxides

Table 4. Cell Stress Effectiveness

Device	τ_{ox}	Normal Operating E-Field	Cell Stress E-Field	Equivalent C.S. Aging Time
2147	700 Å	0.71 MV/cm	1.43 MV/cm 1.0 sec 1.57 MV/cm 1.0 sec	~ 8 hours ~ 80 hours ~ 88 hours
2147H	400 Å	1.25 MV/cm	2.50 MV/cm 1.0 sec 2.75 MV/cm 1.0 sec	2.8×10^4 hours 2.8×10^6 hours ~ 2.8×6 hours

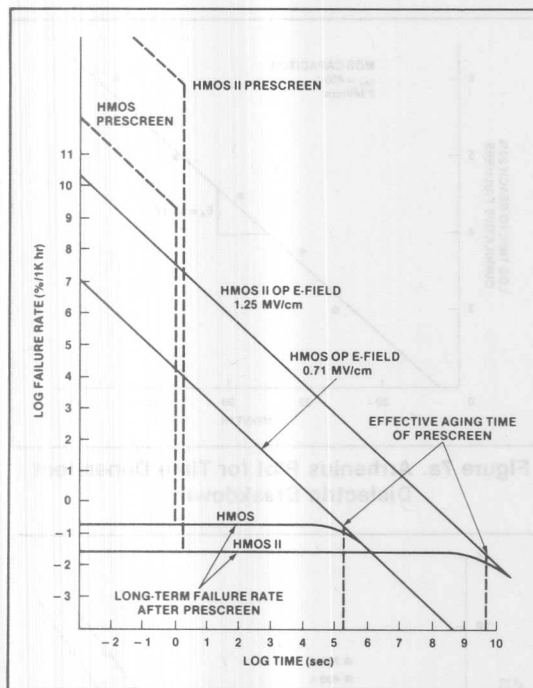


Figure 8. HMOS/HMOS II Theoretical Oxide Failure Rate vs Time With and Without Prescreens

Silicon Defects

Defects in silicon crystal structure have been found to cause degradation failures. These defects are typically inherent in the starting materials, but may also be caused by stresses on the lattice during MOS processing. Silicon defects take the form of increased parasitic leakage. Defects in the channel regions produce leaky transistors, while defects in isolation regions create leakage between devices.

Temperature activation studies indicate this failure mechanism has an activation energy of 0.3 eV to 0.5 eV. These data are presented in Figures 9 and 10. A conserv-

ative 0.3 eV activation energy is used in this report to calculate failure rates. By subjecting these defects to a high voltage stress, as discussed in Intel Reliability Report 7⁽³⁾, severe degradation can be caused in 1-3 seconds. Functional testing after this stress will detect failing devices.

Additional data will be presented in this report which will show that the failure rate contribution of silicon defects is $\leq 0.01\%/1000$ hours (55 °C, 60% UCL).

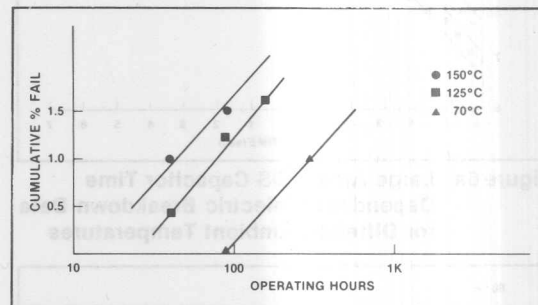


Figure 9. Dynamic Lifetest Results on Non-Prescreened 2147. Shown Cumulative % Fail vs Time at Three Different Temperatures

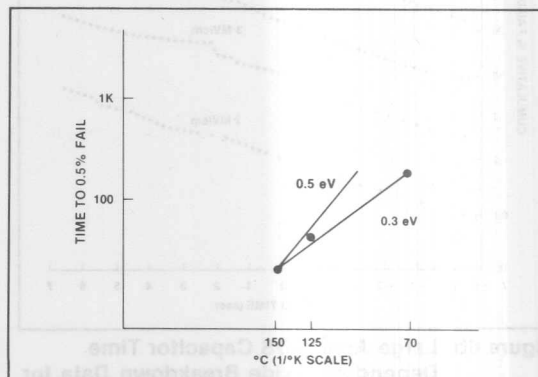


Figure 10. Arrhenius Relationship of Data in Figure 9

Hot Electron Injection

Hot electron injection into the gate oxide near the drain produces a shift in device I-V characteristics. This phenomenon is caused by the generation of electronhole pairs from impact ionization in the high field region near the drain. Electrons with enough energy can be injected into the gate oxide with a small fraction trapped in the oxide causing a negative charge to buildup. This charge buildup results in an increase in the device threshold voltage and severe changes in I-V characteristics, especially when the source and drain are interchanged.

A high voltage, low temperature stress is typically used to accelerate hot electron injection. High voltage accelerates hot electron injection due to increased current densities and electric fields. Low temperatures are used to increase injection currents, prevent punchthrough at high voltage, and minimize photon scattering.

Recent experimentally determined source-drain voltage acceleration, as shown in Figure 11, is in agreement with previously reported data.⁽⁶⁾ The gate-source voltage acceleration factor, however, was found to be non-

existent. The source-drain exponential acceleration factor for HMOS II transistors is:

$$\text{Source-drain acceleration factor} \\ 10^{1/7.7 \times 10^{-3} \text{ MV/cm}}$$

Actual stresses on HMOS II transistors predict ≤ 10 mV shift through 20 years at $V_{GS} = V_{DS} = 5.25\text{V}$. Circuit simulations show that threshold shifts ≥ 100 mV are needed to increase device access time ≥ 1 ns.

In actual memory devices, hot electrons can only occur during transitions (i.e., when $V_{GS} \geq V_T$, V_{DS} decreases), consequently the severity of the problem is greatly reduced. High voltage dynamic lifetests at -70°C , as shown later in this report, demonstrate HMOS II static RAMs to have less than 1 ns access time shift after 1000 hours which is equivalent to 15 years at nominal conditions.

RELIABILITY TESTING

Before introduction of new technologies, Intel's internal reliability goals must be attained. These reliability qualification goals are shown in Table 5. Seven categories of testing are used in the qualification program to assure that the electrical reliability of HMOS II static RAMs meet Intel's reliability goals.

- 1) High Temp Dynamic Burn-in
- 2) High Temp Dynamic Lifetest
- 3) High Temp Reverse Bias
- 4) Low Temp Dynamic Lifetest
- 5) High Temp Storage
- 6) Device Stability
- 7) Temp Cycling

Table 5. Intel Reliability Goals

Infant Mortality	<0.2% after 48-hour @ 125°C
Random	<0.05%/1K hours @ 70°C
Wearout	≥ 20 Years

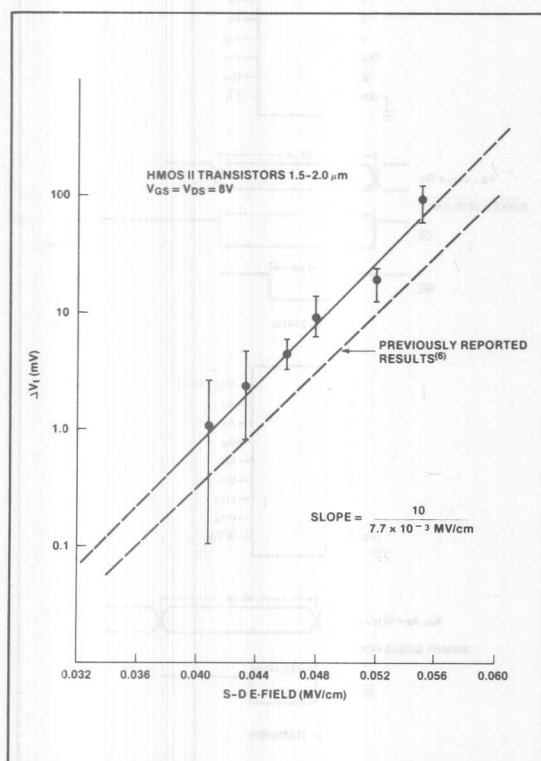


Figure 11. Change in Threshold vs Source-Drain Electric Field During 500 Hour -70°C Lifetest

High Temperature Burn-in

This test is used to establish infant mortality failure rates. Intel defines infant mortality as the early life failures observed after a 48-hour 125°C dynamic burn-in. During the test the memory is sequentially addressed and filled with alternating patterns of ones and zeros. Figure 12 shows timing and connection diagrams for dynamic burn-in on HMOS II static RAMs. In order to eliminate infant mortality fallout in determining long-term failure rates, all devices used for lifetesting are subjected to standard Intel production screens plus a 48-hour burn-in.

High Temperature Dynamic Lifetest

This test is used to accelerate failure mechanisms by operating the devices at an elevated temperature. For static RAMs the operating temperature is 125 °C. The data obtained are translated to a lower temperature using the Arrhenius Plot in Figure 13 giving a larger number of equivalent hours of test.

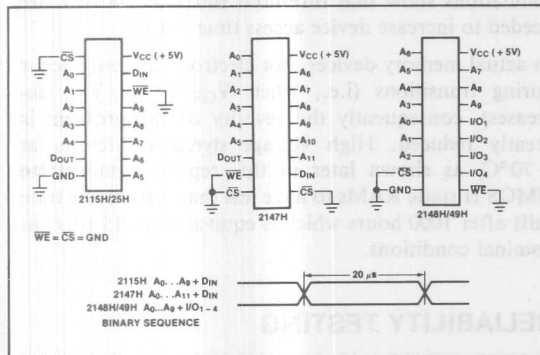


Figure 12. HMOS II Static RAM Burn-in Bias and Timing Configurations

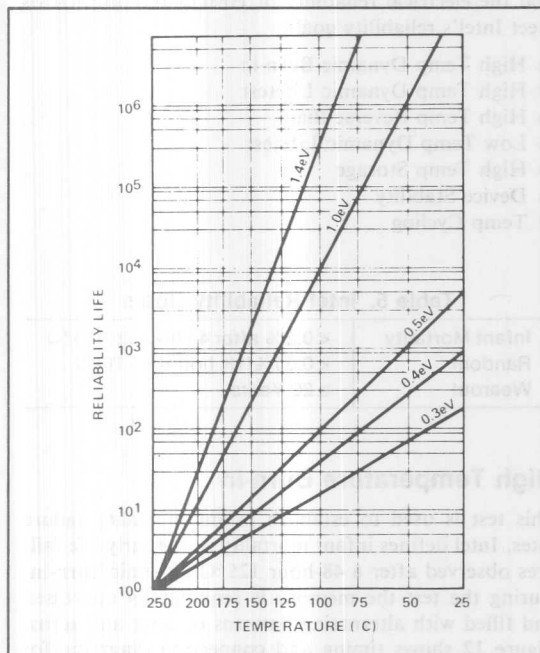


Figure 13. Arrhenius Plot, Which Assumes a Failure Rate Proportional to $\exp(-E_A/kT)$ Where E_A is the Activation Energy for the Particular Failure Mechanism

Dynamic lifetest bias and timing conditions are similar to the burn-in conditions as shown in Figure 14.

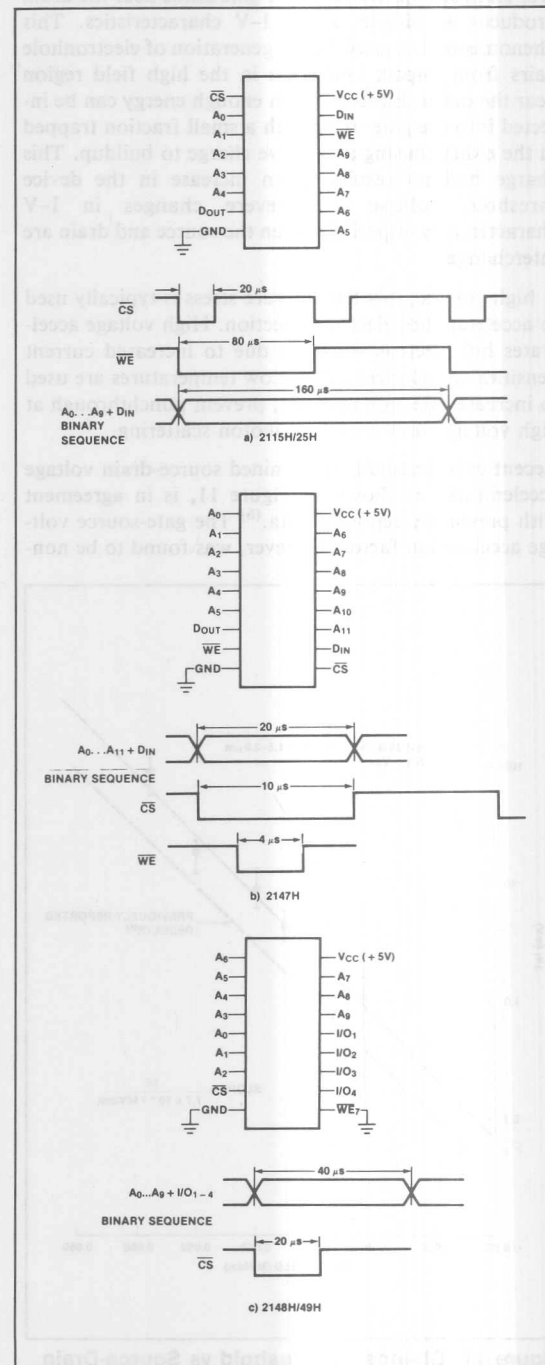


Figure 14. HMOS II Static RAM Dynamic Lifetest Bias and Timing Configurations

High Temperature Reverse Bias (HTRB)

This test is performed to detect failure mechanisms (see Table 3) which are accelerated by high temperature (150°C). This test is effective in accelerating leakage-related failures and drifts in device parameters due to process instability. The HTRB bias diagrams for HMOS II static RAMs are shown in Figure 15.

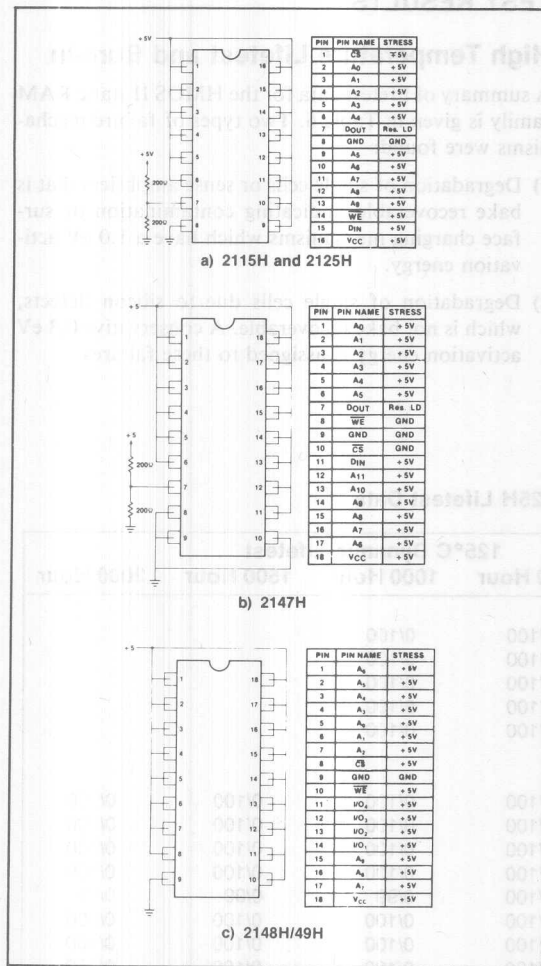


Figure 15. HMOS II Static RAM HTRB Bias Diagrams

Low Temperature Lifetest

This test is performed at maximum operating frequency to detect the effects of electron injection into the gate oxide. The conditions for electron injection occur during transistions when the transistors are in saturation. This test is performed at -70°C in a bath of Fluorinert FC-72 to obtain maximum cooling. Higher than normal

power supply voltage can be used to accelerate this effect. A lifetest at $V_{CC} = 7V$ increases hot electron injection currents by a factor of 13.6 over nominal 5.25 operation.⁽⁶⁾ Access time measurements are used to insure that there is no device degradation.

High Temperature Storage

Another common test is high temperature storage in which devices are subjected to elevated temperatures (160°C for plastic packages and 250°C for hermetic packages) with no applied bias. This test is used to detect mechanical reliability problems (e.g., bond integrity), and process stability.

Device Stability

Since the HMOS II static RAM family operates in high speed memory systems with critical timing, the stability of device parameters is essential. To insure there is no drift in critical parameters, lifetests are conducted on single transistors as well as on actual RAMs.

A good measure of wafer technology stability is the threshold voltage of a single MOS transistor. Transistors are available as test structures on every wafer to monitor process parameters. The transistors are assembled and lifetested under the static bias conditions shown in Figure 16.

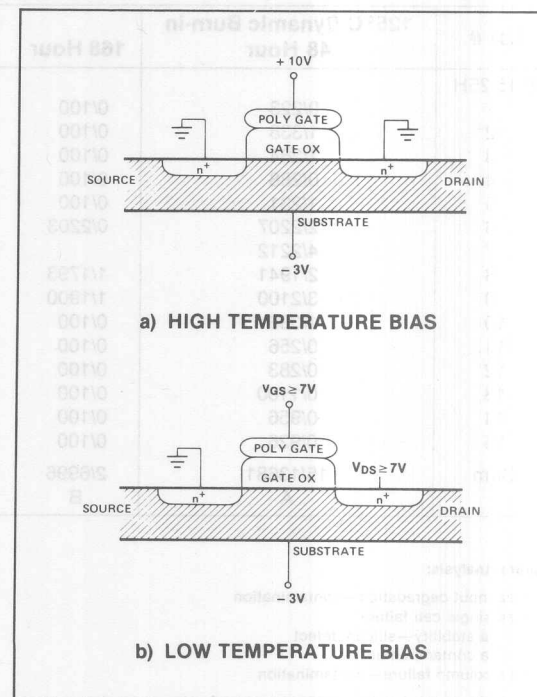


Figure 16. Single Transistor Bias Diagram

This test consists of cycling the temperature of the chamber housing the devices from -65°C to 150°C and is used to detect mechanical reliability problems and microcracks.

Reliability Monitors

In addition to its qualification program, Intel employs a stringent monitor program to assure that reliability goals are maintained. Tests performed to monitor failure rates of current production material are:

- 1) High Temperature Burn-in
- 2) High Temperature Lifetest
- 3) High Temperature Storage

Samples from production material are monitored weekly. Typically a monitor sample of 1000 units are subjected to a 48-hour 125°C dynamic burn-in, then 100 of these units are subjected to a 1000-hour 125°C lifetest

the failure level exceeds Intel reliability goals as previously discussed, a production burn-in sufficient to reduce the failure rate to an acceptable level is implemented or production shipments are held until other corrective actions can be taken.

TEST RESULTS

High Temperature Lifetest and Burn-in

A summary of lifetest data for the HMOS II static RAM family is given in Table 6. Two types of failure mechanisms were found:

- 1) Degradation of single cells or sense amplifiers that is bake recoverable, indicating contamination or surface charging mechanisms which have a 1.0 eV activation energy.
- 2) Degradation of single cells due to silicon defects, which is not bake recoverable. A conservative 0.3 eV activation energy is assigned to these failures.

Table 6a. HMOS II 2115/25H Lifetest Data

Lot #	125°C Dynamic Burn-in 48 Hour	125°C Dynamic Lifetest				
		168 Hour	500 Hour	1000 Hour	1500 Hour	2000 Hour
2115/25H						
1	0/223	0/100	0/100	0/100		
2	1/338	0/100	0/100	0/100		
3	1/284	0/100	0/100	0/100		
4	0/288	0/100	0/100	0/100		
5	1/231	0/100	0/100	0/100		
6	2/2207	0/2203				
7	4/2212					
8	2/1941	1/1793	0/100	0/100	0/100	0/100
9	3/2100	1/1900	0/100	0/100	0/100	0/100
10	1/336	0/100	0/100	0/100	0/100	0/100
11	0/256	0/100	0/100	0/100	0/100	0/100
12	0/283	0/100	1/100	0/99	0/99	0/99
13	0/1100	0/100	0/100	0/100	0/100	0/100
14	0/956	0/100	0/100	0/100	0/100	0/100
15	0/926	0/100	0/100	0/100	0/100	0/100
Cum	15/13681 A	2/6996 B	1/1300 C	0/1299	0/799	0/799

Failure Analysis:

- A) 8 ea input degradation—contamination
 - 6 ea single cell failure
 - 4 ea stability—silicon defect
 - 2 ea contamination
- 1 ea column failure—contamination
- B) 2 ea single cell failure—contamination 1.0 eV
- C) 1 ea single cell failure—contamination 1.0 eV

Table 6b. 2147H Lifetest Data

Lot #	125°C Dynamic Burn-in 48 Hour	125°C Dynamic Lifetest				
		168 Hour	500 Hour	1000 Hour	1500 Hour	2000 Hour
2147H						
1	0/488	0/426	0/200	0/200	0/188	0/188
2	0/281	0/100	0/100	0/100	0/78	0/78
3	0/264	1/100	0/99	0/99	0/99	0/99
4	6/1928	3/1922	0/100	0/100	0/100	0/100
5	0/775	0/775				
6	0/1338					
7	0/1373					
8	0/240	0/100	0/100	0/100	0/100	
9	0/219	0/100	0/100	0/100	0/100	
10	1/277	0/100	0/100	0/100	0/100	
11	2/1079	1/583	0/250	0/100	0/100	
12	1/1064	0/583	0/250	0/100	0/100	
13	2/930	0/583	0/250	0/100	0/100	
14	1/1387					
15	0/200					
16	2/747					
17	0/788					
Cum	15/13378 A	5/5372 B	0/1549	0/1099	0/1065	0/465

Failure Analysis:

A) 13 ea single cell failures

- 5 ea silicon defect
- 8 ea contamination
- 1 ea row failure—contamination
- 1 ea multi-cell failure—contamination

B) 4 ea row failure

- 3 ea contamination 1.0 eV
- 1 ea silicon defect 0.3 eV
- 1 ea single cell-silicon defect 0.3 eV

Table 6c. 2148/49H Lifetest Data

Lot #	125°C Dynamic Burn-in 48 Hour	125°C Dynamic Lifetest		
		168 Hour	500 Hour	1000 Hour
1 2148H	2/175	0/100	0/100	
1 2149H	1/240	0/100	0/100	0/100
2 2149H	0/140	0/100	0/100	1/100
3 2149H	0/250	0/100	0/100	1/100
4 2148H	0/1000	0/100	0/100	0/100
5 2149H	1/1000	0/100	0/100	0/100
6 2149H	0/1000			
Total	4/3805 (A)	0/600	0/600	2/500 (B)

Failure Analysis:

A) 3 ea single cell failure

- 1 ea contamination
- 2 ea silicon defect
- 1 ea multi-cell; contamination

B) 1 ea single cell; contamination (1.0 eV)

- 1 ea multi-cell; contamination (1.0 eV)

Failure rate calculations are shown in Table 7 for each device type. Failure rate calculations are made independently for each failure mechanism. Using the appropriate activation energy shown in Table 3 and the Arrhenius plot in Figure 12, the total equivalent device hours at a given temperature can be determined for each activation energy. The failure rate is then calculated by dividing the number of failures by the equivalent device hours and is expressed as a %/1000 hours. The failure rate is adjusted by a factor related to the number of device hours to arrive at a confidence-level-associated failure rate.⁽⁷⁾ The total device failure rate is then the sum of failure rates of all activation energies.

Combining device hours of the 2115H/25H, 2147H and 2148H/49H gives the highest number of device hours and consequently a more realistic reliability number for the overall technology. This calculation, as shown in Table 7, predicts a failure rate of 0.014%/1000 hours at 70°C and 0.009%/1000 hours at 55°C at the 60% upper confidence level for HMOS II static RAMs.

High Temperature Reverse Bias

The results of high temperature reverse bias lifetesting are shown in Table 8. One failure was observed after 1000 hours at 150°C from 775 HMOS II static RAMS.

Table 8. 150°C HTRB Data

Device	500 Hour	1000 Hour
Cum 2115/25H	0/250	0/250
Cum 2147H	0/375	0/375
Cum 2148/49H	0/150	1/150
Total	0/775	1/775*

*1 ea multi-cell; contamination (1.0 eV)

Low Temperature Device Stability

The results of low temperature lifetesting are presented in Table 9. Included in these data are results of access time stability measurements. The 1000-hour, -70°C

Table 7. Failure Rate Calculations

Device	Device Hours	Activation Energy E _A	Equivalent Hours		# Fail**	Failure Rate	
			70°C	55°C		70°C	55°C
2115/25H	3.05 × 10 ⁶	0.3 eV	1.23 × 10 ⁷	1.97 × 10 ⁷	0	0.008%	0.005%
		1.0 eV	3.26 × 10 ⁸	1.53 × 10 ⁹	3	0.001	<0.001
					Total	0.009	0.005
2147H	2.73 × 10 ⁶	0.3 eV	1.11 × 10 ⁷	1.76 × 10 ⁷	2	0.028	0.018
		1.0 eV	2.98 × 10 ⁸	1.37 × 10 ⁹	3	0.001	<0.001
					Total	0.029	0.018
2148/49H	5.5 × 10 ⁵	0.3 eV	2.23 × 10 ⁶	3.55 × 10 ⁶	0	*	*
		1.0 eV	5.78 × 10 ⁷	2.75 × 10 ⁸	2	*	*
Combined HMOS II	6.30 × 10 ⁶	0.3 eV	2.56 × 10 ⁷	4.09 × 10 ⁷	2	0.012	0.009
		1.0 eV	6.82 × 10 ⁸	3.18 × 10 ⁹	8	0.002	<0.001
					Total	0.014	0.009

*Insufficient data to calculate failure rates.

**Infant mortality failures, 48 hr. @ 125°C, are not included in long term failure rate calculations.

Table 9. Low Temperature Lifetest Results (1000 Hours, V_{CC} = 7.0V)

Access Time (T _{AA}) Stability						
Device	Temperature	No. of Devices	L	Failures	Avg. Shift in T _{AA} *	Max. Shift in T _{AA} *
2147H	-70°C	225	2μ	0	<1 ns	1 ns

*Shifts in T_{AA} (measured at room temperature with V_{CC} = 4.4V) are relative to control units to account for tester variation. Resolution of measurements were ± 1 ns.

lifetest at a stress voltage of 7V results in an acceleration to 15 years at nominal operating conditions. As shown in Table 9, no failures were observed on a total of 225 HMOS II static RAMs with a maximum observed access time shift of +1 ns.

Low temperature threshold stability data on discrete MOS transistors are shown in Table 10. These data on 1.6 and 3 μ devices predict 10 mV shift on a 2 μ device at 25°C in 20 years.

Table 10. Low Temperature Device Stability

L	# Units	Time	V _{GS}	V _{DS}	T _A	Avg. Shift V _T
3 μ	12	1000 Hr	7V	7V	-20°C	≤ 1 mV*
1.6 μ	20	1000 Hr	7V	7V	-70°C	≤ 10 mV**
1.6 μ	15	1000 Hr	8V	8V	-70°C	≤ 10 mV***

*Test Resolution ± 1 mV

**Test Resolution ± 10 mV

***Test Resolution ± 2 mV

High Temperature Device Stability

Threshold voltage stability measurements were made as discussed previously. Table 11 shows the average and worst case V_{TH} shift after a 1000-hour 150°C bias on 3 μ and 4 μ gate transistors that are 100 μ wide. The data in Table 11 show a shift in V_{TH} of less than 1% which is typical of a stable process. Using a thermal activation energy for threshold shifts of 1.0 eV, 1000 hours at 150°C is equivalent to 1.1×10^6 hours (100 years) at 70°C.

**Table 11. High Temperature Transistor Stability
V_{TH} Shift After 1000 Hour 160°C Bias**

L	# Devices	Average	Worst Case
3 μ	15	+1.3 mV	4.1 mV
4 μ	17	+1.0 mV	3.4 mV

High Temperature Storage

No failures were observed during high temperature storage testing on 600 HMOS II static RAMs. A summary of results is shown in Table 12.

Table 12. Bake and Temperature Cycle Results

Device	250°C Bake, 500 Hours	Temp. Cycle, 200 Cycles
2115/25H	0/250	0/50
2147H	0/275	0/25
2148/49H	0/75	0/75
Total	0/600	0/150

Temperature Cycling

After cycling 150 devices, 200 cycles each, there were no failures.

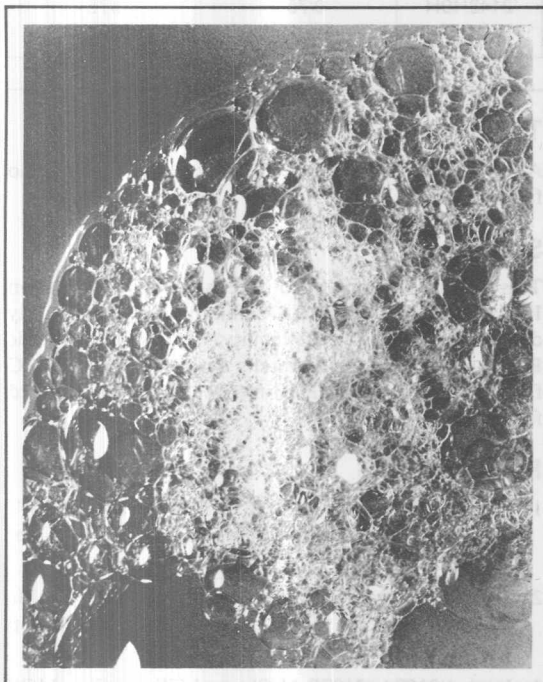
SUMMARY

This report has presented Intel's reliability data on Intel's HMOS II static RAM family. The data clearly demonstrate that the reliability of HMOS II static RAMs meets or exceeds Intel's defined goals. A 55°C failure rate of 0.009%/K hour (60% UCL) is calculated from these data.

REFERENCES

- Jecman, R. M., et. al.; "A 25 ns 4K Static RAM", ISSCC Digest of Technical Papers, pp 100-101, February 1979.
- Jecman, R. M., et. al.; "HMOS II Static RAMs Overtake Bipolar Competition", Electronics, pp 124-128, September 13, 1979.
- Intel, "2107A/2107B N-Channel Silicon Gate MOS 4K RAMs", Reliability Report RR-7, Intel Corporation, September 1975.
- Intel, "2115/2125 N-Channel Silicon Gate MOS 1K Static RAMs", Reliability Report RR-14, Intel Corporation, September 1976.
- Crook, D. L., "Method of Determining Reliability Screens for Time Dependent Dielectric Breakdown", IEEE 1979 Reliability Physics Symposium, pp 1-7.
- Euzent, B. L., "Hot Electron Injection Efficiency in IGFET Structures", IEEE 1977 Reliability Physics Symposium.
- Intel, "8080/8080A Microcomputer", Reliability Report RR-10, Intel Corporation.

A total system solution to magnetic bubble memory applications



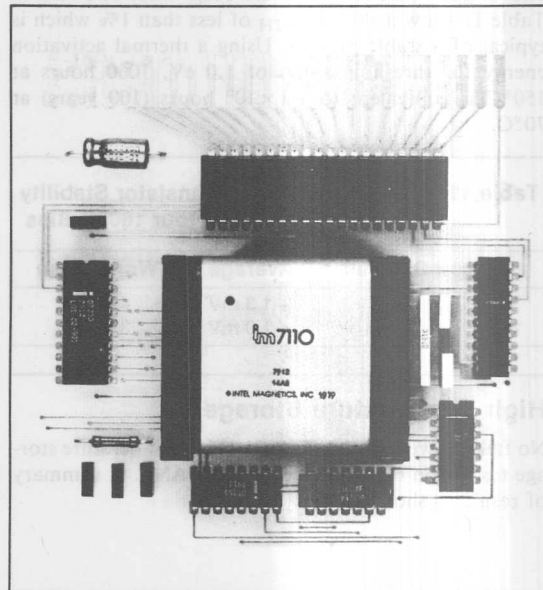
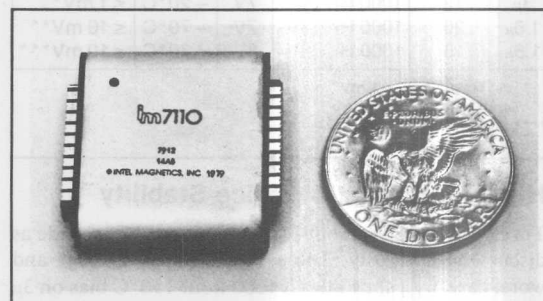
The concept of the magnetic bubble memory (MBM) device is attractive to the system designer because of its high density and non-volatility features. To use the attractive capabilities of the MBM device, the designer must overcome an interface problem more challenging than that posed with semiconductor memories. The designer not only must provide addressing and control logic for the memory device but also must provide precise current pulse generation, low-level analog voltage sensing and relatively high current wave-forms in a set of coil drivers.

Now Intel Magnetics, a subsidiary of Intel Corporation, has introduced the world's first commercial 1-megabit magnetic bubble memory complete with an entire family of support electronics to pose a total system solution for product design.

What is a bubble memory?

It is easy to compare bubble memories to existing semiconductor memories and magnetic storage devices.

Yet, while the magnetic bubble memory device combines many features of existing memories, the important point to focus upon is that the Intel Magnetics 7110 has a unique combination of characteristics that establishes it as a new element in the memory hierarchy. While the bubble memory may compete with specific performance characteristics of existing memories, it will not replace RAMs, ROMs, PROMs, floppy disks or other magnetic auxiliary storage devices. The primary applications of MBMs will be to augment



The Intel Magnetics 7110 (top), is a one million bit bubble memory. The MBM device is supported by a complete family of large scale integrated circuits to provide product development without concern for drive and interface details.

other memories and to develop new products based on its inherent unique characteristics.

Bubble memories are compact, non-volatile mass storage elements processed in a manner similar to silicon wafer fabrication.

Data is stored as magnetic "bubbles" in a very thin film of synthetic garnet. The bubbles are microns in size and move in a plane of the film when a magnetic gradient is present. Viewed under a microscope with linear polarized light, the bubbles appear to be fluid circular areas that step from space to space following fixed loops and tracks. The IM 7110 is organized as a serial-in parallel loop serial-out shift register as shown in Figure 1.

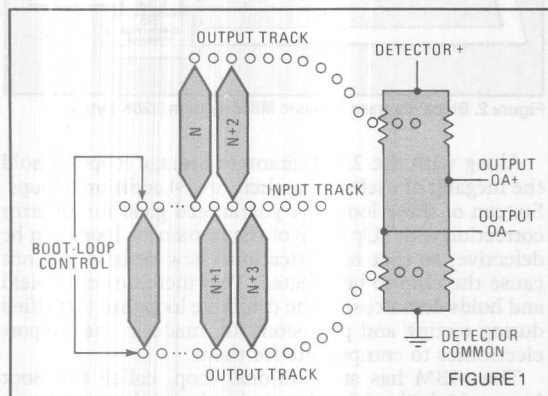
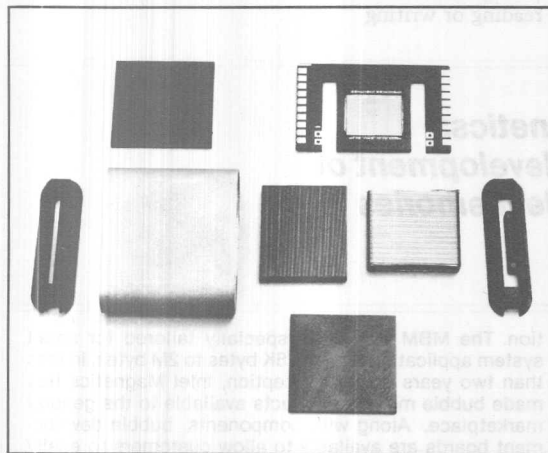


Figure 1. A serial-in parallel loop serial-out shift register. One half of the 7110 is shown. Loops N, N+2, etc. form the even quad of that half and loops N+1, N+3, etc. form the odd quad.



Package for Intel Magnetic 7110 consists of substrate with bubble memory chip, coils, permanent magnets, end caps, and magnetic shield.

The bubbles can be controlled to perform memory functions. Corresponding to the on/off concept for semiconductor memories, the presence of a bubble represents a binary "1" and the absence of a bubble represents a binary "0". The bubbles shift synchronously

around storage loops and along input-output tracks in step with the rotating magnetic field which is in the plane of the wafer. (The film is magnetically polarized in one direction and the bubbles are of reverse magnetic polarity).

Data is read in a unique manner. A bubble from each loop is replicated (duplicated). One of the two bubbles continues around the loop to retain memory while the other proceeds along the output track to the detector.

Standard photolithography is used to make conductor and magnetic permalloy patterns on the chip. A pair of ac-driven crossed wire wound coils are slipped over the chip to provide the rotating magnetic field. The system is stabilized with a pair of permanent magnets and is protected from external magnetic influences by a sleeve of shielding material. The protection allows the device to be used around CRT coils, transformers and other equipment that produces magnetic fields.

Bubble memories compared to other memory devices.

MBMs can storage comparatively huge amounts of information, are non-volatile, compact, highly reliable and can be used in harsh environments.

The 7110 has a normal data capacity of 1,048,576 bits (128K bytes). In comparison, the largest RAM recently announced has 64K bits (8K bytes), the largest announced ROM has 128K bits (16K bytes), and a single-sided minifloppy disk can only store approximately 720,000 (90K bytes). 1M bits of information is over 30 pages of single-spaced typewritten pages of data.

Non-volatility is one of the most important characteristics of the MBM. Data is retained if power is removed as the bubbles stay in position indefinitely. While ROMs and PROMs are also non-volatile, the MBM data can be changed or modified at the same rate as it is read.

Physical size is another significant feature of the MBM approach. In fact, one of the early applications for bubble memories will be to reduce the physical size of equipment. The MBM takes up a lot less physical space than either a tape or a disk. Even when grouped together to form a 1 or 2-megabyte system, bubble memories save considerable board space over semiconductor memories of similar system capacity. The minimum 7110 system (128K bytes) resides on a board size of less than 16 square inches.

As for reliability, bubble memories have a high resistance to shock, vibration, humidity and radiation, making them ideal for use in harsh environments. Extra storage loops are built-in for error correction.

The bubble memory has a slower access speed than the semiconductor memory, but it is faster when compared with other magnetic media. The 7110 has an average access time of 40 milliseconds, or 80 milliseconds in the worst case. Once a block of data is moved into position it comes out at the rate of 68,000 bits per second.

The Intel Magnetics approach to bubble memory

technology goes beyond the mass storage device. As mentioned, the MBM is supported by a family of components created by Intel and shown in the block diagram of Figure 2. The basic components are used to build a minimum system of 128K bytes in a 16-square-inch board. Up to eight MBMs can be interfaced to one controller for a megabyte (8,000,000 bits) of storage for larger systems. The system interfaces directly with the Intel microprocessor bus system through the 7220 Bubble Memory Controller. Therefore, the memory can be treated as a slave to the 8080, 8085, 8086 or 8088 host system.

The key functions of the system include binary data organization, standard +12 volt and +5 volt power supply operation, transparent handling of spare loops, flexible multiple MBM operation, single page (512 bits or 64 bytes) or multiple page data transfers and built-in error correction.

The 7110 has binary page organization.

As mentioned, the 7110 is a serial-in parallel loop serial-out shift register storage device. The device has binary page organization, i.e., it stores 2,048 pages of 512 bits each. The pages are divided into two channels of 256 bits each. There are 128 data storage loops per channel divided into two sections of 64 data loops each. There is a separate detector for each channel. Through one detector, bubbles are shifted every other cycle. With all four quads interleaved, the maximum data rate is twice the shift rate. A page address is selected and shifted to the starting location for a read or write operation, and the bits of a new page are first written serially on an input track. The bits shift until they coincide with

the bits of the page in storage to be replaced. A swap operation then exchanges a new page for the old at the address location selected.

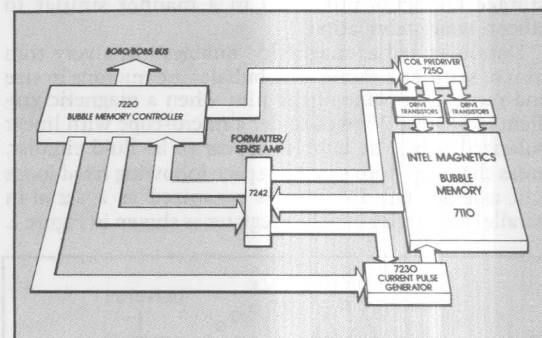


Figure 2. Block diagram of basic MBM system (128K bytes).

Along with the 256 guaranteed storage loops to hold the megabit of useful data, there are 64 additional loops. Sixteen of these loops are guaranteed good for an error correction code. Up to 48 of the remaining loops can be defective, so that in fabrication a few defects will not cause the chip to be rejected. This increases chip yield and holds down costs. The defective loops are identified during testing and provisions are made in the support electronics to compensate for them.

The MBM has an additional loop, called the Boot Loop, which identifies the good and spare loops and contains a loop map. The map is read out of the 7110 MBM and stored in the 7242 Dual Formatter/Sense Amplifier (FSA), each time the system is initialized prior to reading or writing.

Intel Magnetix: Dedicated to the development of magnetic bubble memories

Intel Magnetix, Inc. was organized in October of 1977 as a subsidiary of Intel Corporation and was given the charter to develop and market a family of components for magnetic bubble memory subsystems.

To make this possible, a strong technical team was assembled to produce a one megabit bubble memory part and four LSI circuits, built with four separate Intel silicon processes—bipolar, CMOS, NMOS, and HMOS.

The key design features incorporated in the Intel Magnetix MBM system approach were chosen after a strong effort to define the appropriate market niches for bubble memories. Ideas from more than two hundred potential customers were gathered, which greatly influenced the choices made for size, speed, and organiza-

tion. The MBM system is specially tailored for small system applications from 128K bytes to 2M bytes. In less than two years from its inception, Intel Magnetix has made bubble memory products available to the general marketplace. Along with components, bubble development boards are available to allow customers to easily begin an evaluation of the IM bubble memory product line.

Intel Magnetix has maximized the potential for bubble memory today by offering the first practical one megabit bubble memory component and the first family of semiconductor support circuits that combines minimum part count with features like parallel/multiplex operation and built-in error correction.

In operation, one page or a group of pages can be read or written for a given system request. Upon completion, the bubble device itself can be stopped until the next request. This start/stop feature can reduce the average page access time in systems where successive page accesses are not random. Least recently used (LRU) and look-ahead algorithms can be used to put expected future pages at the locations corresponding to the start of the page read or write cycles.

TABLE 1. BUBBLE MEMORY SYSTEM PERFORMANCE

	One MBM	Four MBM's	Eight MBM's operated in parallel	Eight MBM's multiplexed one at a time
Capacity	128K Bytes	512K Bytes	1 Megabyte	1 Megabyte
Nominal Data Rate	68 KHz to 136 KHz	272 KHz to 544 KHz	544 KHz to 1088 KHz	68 KHz to 136 KHz
Avg. Access Time	40 to 20 ms	40 to 20 ms	40 to 20 ms	40 to 20 ms
Power Dissipation (100% duty factor)	6 W	20 W	40 W	11 W
Standby Power	1.3 W	3.7 W	7.0 W	7.0 W
Board Area	16 sq. in.	45 sq. in.	90 sq. in.	90 sq. in.

At the systems level, data rate can be increased by operating bubble devices in parallel. The 7220 controller can handle eight MBMs in parallel. Using eight of the 7110, 50KHz devices, the nominal bit rate becomes 544 KHz compared to 68KHz for a single device. Using 16 devices connected to two controllers, the bit rate is 1.088 MHz. Table 1 shows the expected performance characteristics of the 7110.

Support electronics complete the system.

Support electronics for the MBM system are shown in the basic system block diagram, (Figure 2).

User interface is provided by the 7220 Bubble Memory Controller (BMC). The BMC is a 40-pin device built with HMOS technology. It provides bus interface, generates all memory system timing and control, maintains memory address information, and interprets and executes user requests for data transfers. From a practical standpoint, the 7220 interface makes the MBM system look like a peripheral to the microprocessor system bus.

The 7242 Formatter/Sense Amplifier (FSA) is actually a dual channel unit to interface with both channels of the bubble memory. It is a 20-pin device built with NMOS technology. It senses the low level bubble signals, handles redundant loops and buffers data. It also contains the burst error detection and correction circuits for each channel.

The 22-pin Schottky bipolar 7230 Current Pulse Generator (CPG) supplies the peak currents required by the MBM. It also contains a power down circuit to shut off the current sources whenever the device is

deselected and it has power failure detect circuitry to shut off pulses to the bubble memory.

Relatively high peak currents are required to drive the coils. Therefore the 7250 Coil Predriver (CPD) interfaces the 7220 BMC to driver transistors which can be quad bipolar transistor packs or Intel's 7254 quad VMOS FET transistor packs. This CMOS device is supplied in a 16-pin DIP package.

The significance of these support circuits is that they provide all the complex control and interface necessary between the system bus and the MBM in a simple, flexible manner. The user can start with a compact development board or connect the components of his system board with a minimum of effort. These LSI circuits replace what would otherwise be a board full of control electronics. They make it practical for the OEM to use the 7110 MBM in production products.

The MBM system provides flexible system organization and a variety of data transfer rate methods.

With the support electronics of the MBM system, the design timing problems are reduced to interfacing to a standard bus. The system operates on +12V and +5V only and contains monitoring circuitry. If voltages drop below acceptable levels, the system goes into an orderly shutdown, preserving data.

The support electronics are widely flexible, providing for different system designs.

A single 7220 BMC can directly control up to eight MBMs. Each cell consists of a bubble memory device, a 7230 CPG, a 7242 FSA, a 7250 CPD and two quad transistor packages.

Further expansion can be accomplished in two ways. First, provisions are made in the BMCs for paralleling controllers. This provides a wider word width at the bus and still allows the controller to accommodate up to eight memory devices. For example, two BMCs operated in parallel with sixteen I/Os results in a 16-bit wide word and 2 megabytes of capacity. For the second approach, since each support device has a chip select pin, banks of MBMs can be switched into or out of the circuit under external control.

Even where eight MBMs are used within a system, the user can still access data from a single bubble memory device. The controller transfers data and commands to the FSA through a serial bus. The controller uses a time-division multiplexing scheme to allow individual FSA channel addressing, (remember the 7242 contains two channels). To communicate with the FSA—the heart of the system—the controller outputs a sync pulse. Simultaneously, a data stream is output on the serial bus. As each FSA receives its sync pulse, it examines the serial bus to determine whether or not it is being addressed. Commands are distinguished from data by a C/D (Command/Data) pin controlled by the 7220 BMC. Any FSA channel that is not addressed automatically deselects itself and removes itself from the bus. Each FSA channel is active only when selected and

The FSA creates system flexibility.

One of the key functions of the 7110 is the flexible multiple MBM operation. In any configuration, data is transferred in serial form and is reassembled into an

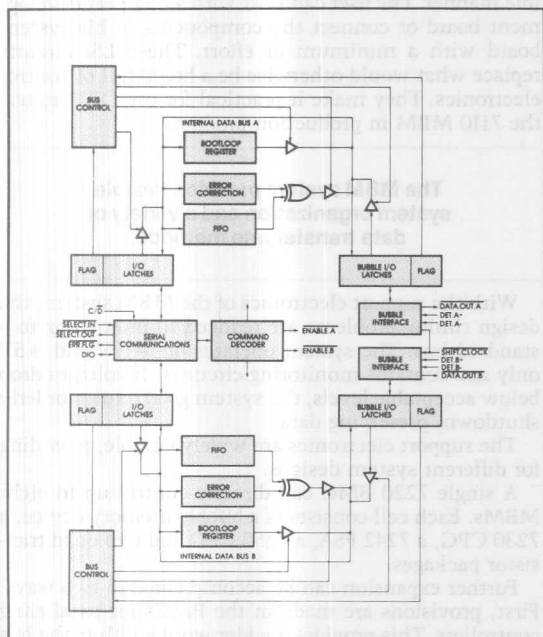


Figure 3. 7242 Formatter/Sense Amplifier logic diagram.

8-bit byte by the controller. The serial bus operates at a minimum rate equal to 20 times the rotating field rate so that a bit of data is transferred to or from each FSA channel during each bubble field rotation. Therefore, a system containing a megabyte of bubble memory can be useful in low data rate, low power systems as well as in high performance systems. The partitioning of the data handling into the FSAs is what allows this flexibility.

The FSA enables the designer to perform a function previously missing in MBM systems—simplified paralleling and multiplexing of memory devices. This is accomplished with a minimum of parts as the sense amp and redundancy handling (of the defective loops) are combined in the same package.

A map of the defective loops is contained in the additional bootloop on each MBM chip. The map is read and stored in the FSA during system initialization. The controller sees only good data bits and does not have to be concerned with maintaining a map for bad loops for each MBM.

The FSA also handles error correction and detection.

if errors are present, correct (within the capabilities of the code) the data before it is transferred to the host.

Since the majority of the errors are likely to be detection or read errors rather than data (bubble loss) errors, repeating reads can generally be avoided.

The user can also use his own error correction code. An additional 16 loops are available for user-implemented error code, additional data storage or page address header storage.

The controller provides a variety of data transfer methods.

The 7220 controller provides for the user three methods of data transfer across the system bus; interrupt driven I/O, polled I/O and Direct Memory Access (DMA).

The first two methods require that data be transferred to a register in the host CPU prior to being stored in memory. In DMA, data passes directly to the host memory. High performance systems that use parallel bubble devices must choose the DMA method as the data rate can exceed the I/O capabilities of most microprocessors. Eight parallel 7110 devices can achieve a maximum data transfer rate of 100K byte/sec (10

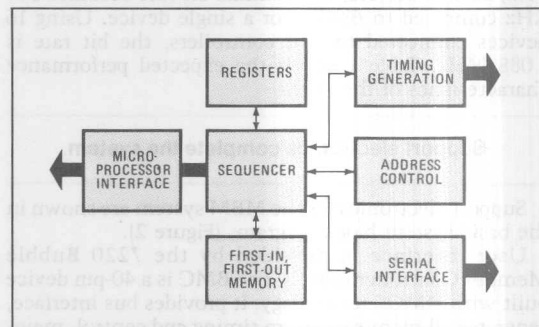


Figure 4. 7220 controller block diagram.

$\mu\text{sec}/\text{byte}$). A typical microprocessor instruction execution requires $2 \mu\text{sec}$. Transfer under CPU control would require several instructions.

The 7220 controller is designed to interface directly to an 8257 DMA chip. Where DMA is not desired, the DMA request pin can be used as an interrupt to signal the host that data is available. The DMA request pin is set whenever the controller FIFO is half full (during a read) or half empty (during a write). This pin can be used as a data pin, or as a second level of interrupt that guarantees the host can read or write a minimum number of bytes (20) to the controller.

The lowest performance method—polled I/O—is provided by a status bit in the controller that indicates

presence of data in the FIFO. The host could continually poll status and output data when FIFO is available, but this is useful only in the lowest performance system.

The operation of the system is straightforward.

After power up, the host must initialize the system. Communication with the controller is accomplished via a set of addressable registers contained within the controller. These registers are addressed by the host and written with the data transfer method information. The host then issues an initialization command and the controller proceeds to read the boot loop of each bubble device, writing the bad loop information into the boot loop register contained in the corresponding FSA. The

controller begins with the first FSA in the chain and continues in order until all devices present have been initialized.

After initialization, the controller will interrupt (if enabled) the host and is ready for data transfers. To initiate a transfer, the host writes an address into the controller registers and issues a read or write command. The controller then accesses the desired page(s) of information and performs the operation. An interrupt is normally issued upon completion, but it may be inhibited by the user. An address is maintained and updated for each MBM in the system within the controller. If a block transfer overflows the address boundary of the bubble device, the controller automatically switches to the next device in the chain. A maximum of 2048 pages of data can be transferred with a single command se-

Applications for bubble memories

Initially, bubble memories are expected to be used in microprocessor applications requiring 128K to 2-megabytes of storage. Such current applications include terminals, word processing systems, telecommunications and process control applications. Basically, they will be used wherever non-volatile program or data storage is required.



Non-volatile magnetic memory has many advantages for a multi-terminal system.

Currently, bubble memories will provide very large amounts of non-volatile solid-state memory that can be modified. They will not replace any existing forms of memory but will augment other memory devices and may be used to develop new products.

One of the first applications for the bubble memory is to reduce equipment size. As the MBM can store a tremendous amount of data in a very small area, the possibilities to make equipment more compact are increased. This leads to more portability of existing products, to new end-use products and to new markets for microprocessor based systems. In the next five years, higher density and lower cost will allow the bubble

memories to be used in minicomputers and large computer systems. They will act as fast cache or buffer memories for even larger mass memory units.

The MBM can be treated as a "sometimes changed" ROM or PROM. As an example, if an MBM is resident in a terminal connected to a larger system, either permanently or through a modem, the system can, from time to time, modify the program such as updating a price or tax table stored in the bubble memory. The MBM can also be treated as a PROM with the program changed in the field or by a device change.

As a more "frequently changed device" the MBM can be treated as a RAM. The MBM can be used in a terminal to keep track of inventory and sales figures. The figures are transmitted to a central computer at night when transmission costs are low. As bubble memory speeds are compatible in applications involving data with human interface, data might be read in and out while programs are transferred to RAM for faster execution.

Bubble memories offer a low cost, light-weight alternative to RAM with battery backup where non-volatility is essential.

MBMs can augment disk products to reduce many electromechanical service and maintenance headaches. Compared to tape or disk where dirt, dust and handling are frequent problems, the bubble's magnetic film is free from contamination. Therefore, MBMs can be produced for applications in garages, machine shops and grocery stores as well as offices and computer rooms.

The MBM can be used as a second disk for a system. The first disk loads the programs which are then transferred to the MBM. The bubble memory provides more reliable operation with faster access times. A similar system would be implemented with programs entered via tape.

In the future, the magnetic bubble memory will be used for diverse applications ranging from low-performance terminals to high-performance mass storage systems.

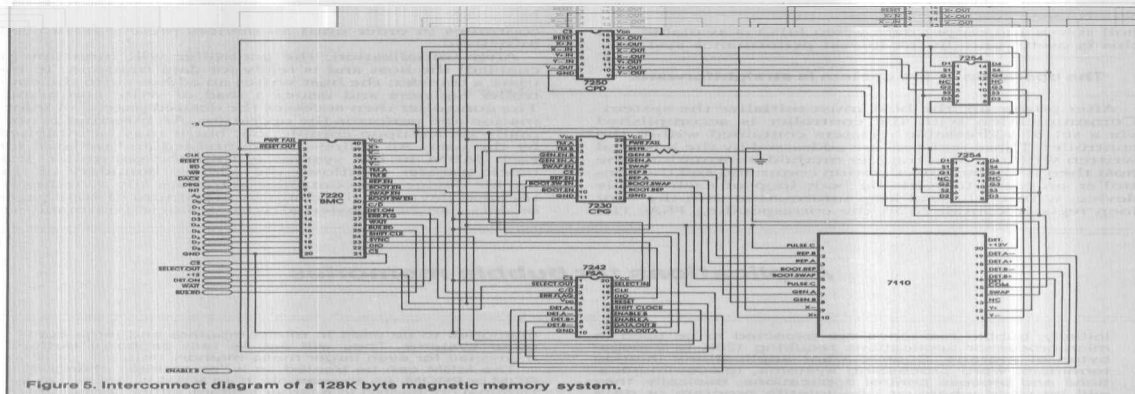


Figure 5. Interconnect diagram of a 128K byte magnetic memory system.

quence. If a power down occurs during a transfer, the controller automatically shuts down the coil drivers in the proper sequence (providing power remains at acceptable levels for about 100 microseconds).

The user requests data transfers by addressing a command/status register in the 7220 BMC. A large number of commands are available to the user, including several that are useful for system diagnostics.

The most commonly used commands are:

- (1) Initialize—performed after power up to reset the system
- (2) Read—causes the selected pages of bubble memory to be accessed
- (3) Write—writes user data into selected bubble memory pages

Read and write can be specified for up to 2048 pages. In addition, a seek command is provided for the user who can predict the next read address. This avoids an additional latency. However, in multiple page reads (or writes), consecutive page addresses are physically located such that the next page is available immediately after completion of the preceding read or write. Additional commands include reading and writing of the boot loop registers in the FSAs, or the boot loop on the bubble chip, a software reset, and an abort command.

Status bits provided to the user include a busy signal, an operation complete flag, a FIFO ready flag, and several error flags including timing error, correctable error, and uncorrectable error. Status of each FSA can be

determined by the host processor by means of a special command. Interrupt masking is provided so that the user can decide whether he is interrupted by errors as well as normal (operation complete) interrupts. The combination of error correction capability, a versatile command set and appropriate status flags allows the user to perform on-line maintenance checks to enhance system reliability.

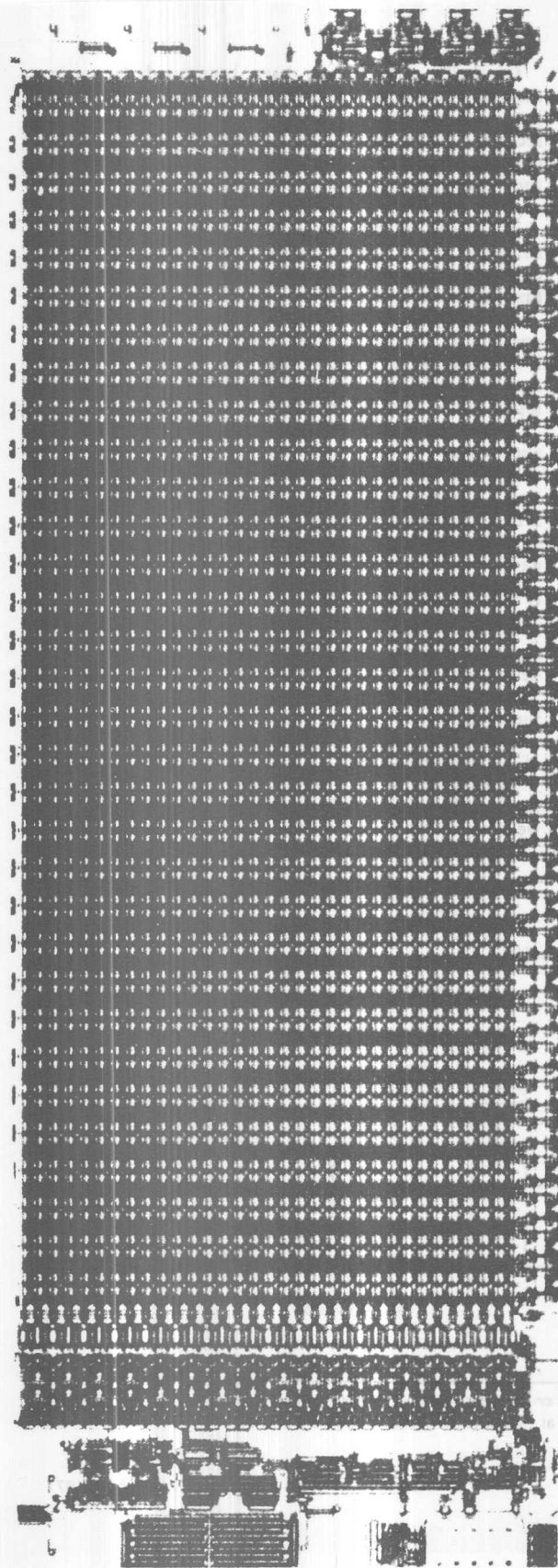
A complete system interconnect diagram for a 128K byte system is shown in Figure 5. With the system support and interface circuits available from Intel, designers can now treat bubble memory as a unique new solution to mass storage applications.

AVAILABILITY: Now
LITERATURE: Bubble Memory Design Handbook N/C

Article Reprints

5

A grid of 10 columns and 10 rows of squares. The bottom row is significantly larger than the others, while the other rows are of uniform height. The grid is composed of 10 columns and 10 rows of squares. The bottom row is significantly larger than the others, while the other rows are of uniform height. The grid is composed of 10 columns and 10 rows of squares. The bottom row is significantly larger than the others, while the other rows are of uniform height.



Speedy RAM runs cool with power-down circuitry

Static RAM's low-power standby mode minimizes memory system dissipation

by Richard Pashley, William Owen, Kim Kokkonen, and Anne Ebel, *Intel Corp., Santa Clara, Calif.*

□ A new form of computer data storage—fast main-frame memory—is heralded by a new high-density, fully static random-access-memory chip that blends high speed at the chip level with low power dissipation at the system level. This marriage is accomplished in the 2147, a 4,096-bit static RAM that combines a high-performance metal-oxide-semiconductor technology (H-MOS) with circuit innovation to attain a new power-down mode.

The H-MOS process gives the 2147 access and cycle times competitive with bipolar technology—typically 45 nanoseconds—and superior speed-power performance. But the real key to the RAM's practical use in large, fast memory systems is its unique power-down capability.

Raw speed has always been restricted to use in scratchpads and other small memory systems, where cooling problems are not severe. However, building compact main memories and cache memories into computer mainframes requires modules containing high-density RAM arrays and, often, closely stacked memory boards. The 2147 provides a new way of minimizing power dissipation that makes this construction practical. It also will further simplify design of small systems.

The 2147 goes on standby automatically when the chip is deselected. Its typical power dissipation drops from 500 milliwatts to only 50 mw. More important, for a rock-solid memory design, worst-case dissipation drops from 880 mw to 100 mw, compared with a watt or more of continuous power dissipation for conventional bipolar static RAMs. These power ratings are for the standard 2147, which has a maximum access time of 70 nanoseconds and an identical cycle time. For higher-performance applications, a premium part is offered with a 55-ns worst-case access-cycle-time specification.

There is no access-time penalty for the low-power standby feature. The access time from chip select (power up) is equivalent to the access time from an address transition with the chip previously selected. Chip select has no special timing requirements: it can come up before, after, or coincident with address change.

Since the fraction of the RAMs selected during any

given cycle in a large system can be small, using chip select to control power down and power up gives the system designer a simple means of solving power distribution and cooling problems. That is, modules containing large numbers of RAMs will operate at much lower average power than small modules, and the system designer can easily keep a low power density throughout the system. Equally important, the mode does not sacrifice access time or complicate design.

Breaking with tradition

Bipolar RAMs have dominated high-speed memory design since the dawn of semiconductor memory technology in the 1960s. However, they are costly and very power-hungry, all but ruling out use in large, fast, main-frame memories.

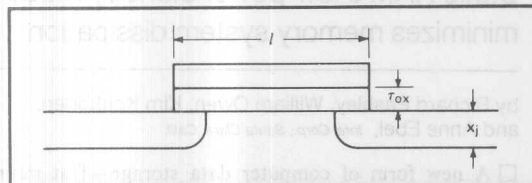


TABLE: MOS TECHNOLOGY EVOLUTION

Parameter	MOS, 1976	H-MOS, 1977
Channel length, l (μm)	6	< 4.0
Gate-oxide thickness, τ_{ox} (\AA)	1,100	< 1,000
Junction depth, x_j (μm)	1.7	1.0
Depletion loads	yes	yes
Oxide isolation	no	yes
Built-in substrate bias	yes	yes
Speed-power product (pJ)	4.0	1.0

Generally, MOS devices have been used for main memory. They enjoy an edge in speed-power products but were unable to approximate bipolar speed until recently. Some early MOS RAMs that attempted to compete with transistor-transistor-logic static RAMs in the under-100-ns market required multiple power supplies and clocked chip-enable operation.

The 2147 offers the speed, compatibility, and simplicity of a static bipolar RAM built with TTL circuitry, yet provides the low power of MOS. It is the first high-speed RAM with this combination of features.

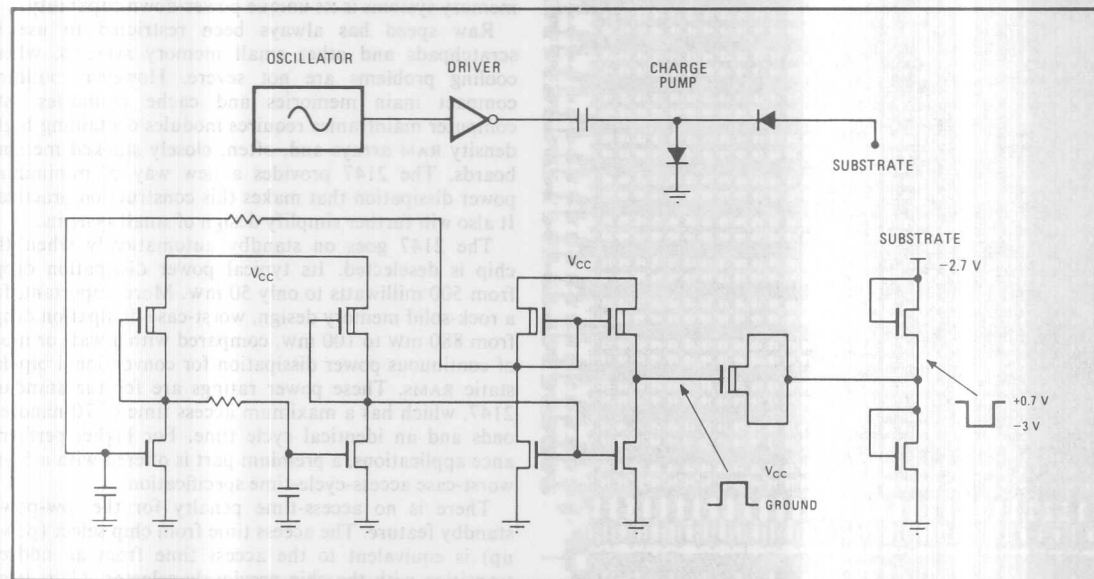
Moreover, the device is extremely easy to use. It operates on a single +5-volt supply like TTL devices and is designed so that a supply with 10% tolerances may be used. Input/output levels are also TTL, and unlatched inputs and outputs ensure simple, static timing.

The output typically sinks 25 milliamperes at 0.45 v and sources 15 mA at 2.4 v—ample current to eliminate output drive and sensing problems. Further, valid operation is guaranteed with input swings as small as 0.8 v to 2.1 v. Finally, the 2147 has an industry-standard 4,096-by-1-bit configuration and is packaged in a standard 18-pin dual in-line package.

H for high performance

H-MOS technology reduces the physical parameters of n-channel, silicon-gate MOS to new lows. It combines device scaling with on-chip substrate bias generation. This results in higher density and a 4:1 improvement in the speed-power product (see table) making the 2147 chip the smallest and fastest of the emerging generation of 4-k MOS static RAMs.

By reducing the physical parameters of the device by a fixed scaling factor, circuit density and performance were increased while active circuit power decreased. In



1. Getting back bias. To minimize the substrate's body effect, this on-chip back-bias circuit has a self-starting oscillator driving a charge pump that is capacitively coupled to the substrate. The oscillator runs at 13 MHz to maximize the pump's efficiency.

the H-MOS process, polysilicon gate lengths have been shortened to less than 4 micrometers and gate-oxide thickness to less than 1,000 angstroms. Using arsenic as the source-drain gives shallow junctions ($<1\mu\text{m}$). Circuit performance and density improve still further with use of oxide-isolation and depletion-load processing. Finally, substrate biasing reduces body effect and parasitic junction capacitance—the back-bias voltage is generated on board to eliminate the requirements for an additional pin and power supply.

As a result of these design factors, the power figure of merit is 1 picojoule (in an 11-stage ring oscillator with a per-stage fan-out of 3). Conventional $6\text{-}\mu\text{m}$ -gate n-channel MOS has a 4-pi speed-power product.

Biasing the chip substrate

As device elements shrink and make substrate effects more noticeable, reverse or back biasing becomes more important for device performance. The 2147's on-chip bias voltage is self-regulating, so it needs no special regulation circuitry. Also, it tracks fluctuations in the 5-v supply, temperature changes, and process variations.

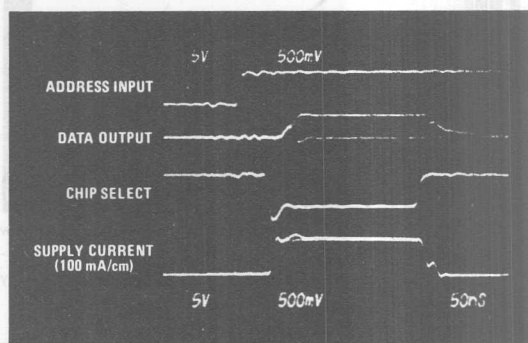
The circuit typically generates -3 v back-bias voltage and consumes 7 mw. It consists of a self-starting oscillator driving a small charge pump that is capacitively coupled to the chip substrate (Fig. 1). Its 13-megahertz frequency was selected to optimize the efficiency of the charge pump. The oscillator is inherently unstable—deliberately not balanced—to assure self-starting under all conditions. The charge pump is small—about the area of two or three bonding pads—since the generator's only current drain is substrate leakage.

The chip's substrate diffusion capacitance is large enough to absorb the effect of momentary substrate-current spikes. Input coupling and internal node

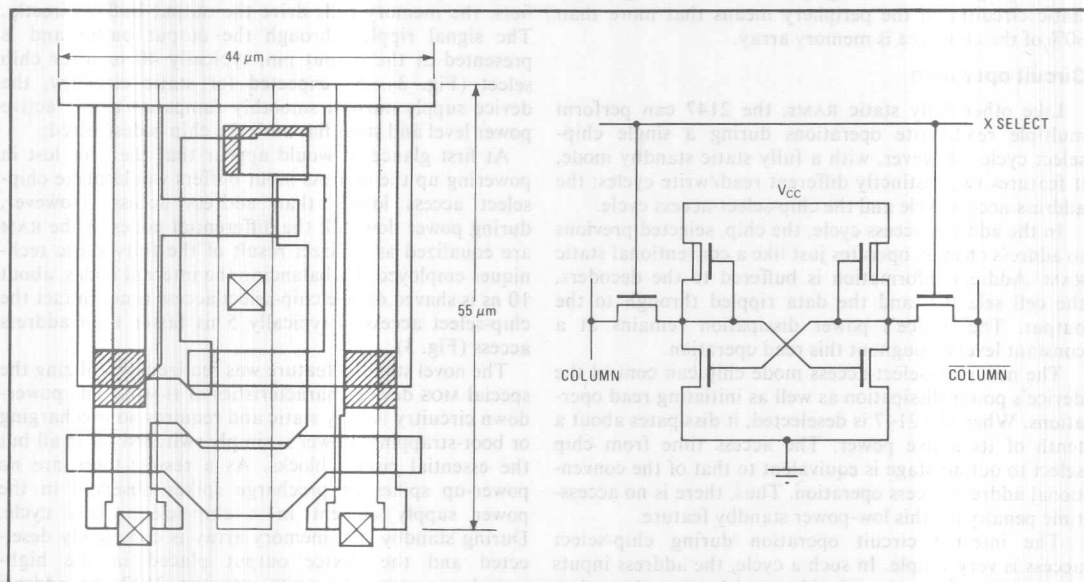
switching during memory accesses cause back-bias differentials of less than 100 millivolts.

A glance at the cell schematic (Fig. 2) indicates that the 2147 is still a fully static RAM like its grandparent, 1,024-bit 2102A. The cell is a conventional, six-transistor, cross-coupled flip-flop that uses depletion-load devices. It occupies only 3.75 square mils. Typically, it dissipates 5 microwatts of power, giving a power dissipation for the full 4-k memory array of 20 mw.

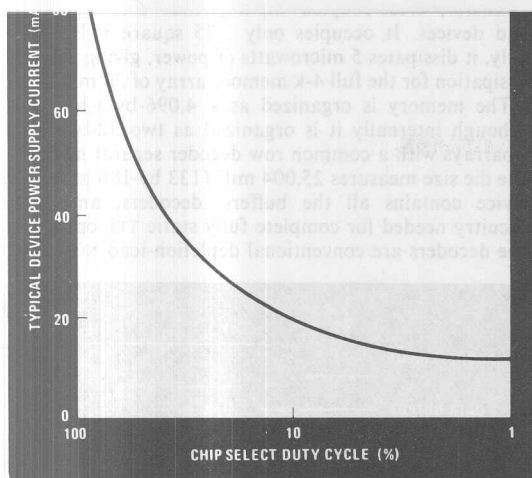
The memory is organized as a 4,096-by-1-bit RAM, although internally it is organized as two 32-by-64-bit subarrays with a common row decoder separating them. The die size measures $25,004\text{ mil}^2$ (133 by 188 mil). The device contains all the buffers, decoders, and write circuitry needed for complete fully static TTL operation. The decoders are conventional depletion-load NOR gates



3. Quick and steady. The 2147 is fast, with the data output beginning to appear 40 ns after the address input goes up. No spikes appear in the supply current trace after power down; power is simply switched off in all but the essential circuit blocks.



2. The cell. While the 2147's cell is a conventional six-transistor design, the innovative H-MOS process reduces its size to half that of ordinary static RAM cells. The cross-coupled flip-flop design lays out in only 3.75 mil² and points to future H-MOS static RAMs of even greater density.



4. Saving energy. For large memory systems that operate at low duty cycles, the 2147 saves power, as the device's typical supply-current characteristic shows. At 10% duty cycle, the part burns only 20% of the supply current it uses at full duty cycles.

with a power-down switch in the power-supply line. The input buffers are similar to those of the 2102A—a simple string of inverter gates driving two push-pull output stages. The RAM requires no clocks or internal precharging to attain its high performance. Using simple static circuitry in the periphery means that more than 60% of the chip area is memory array.

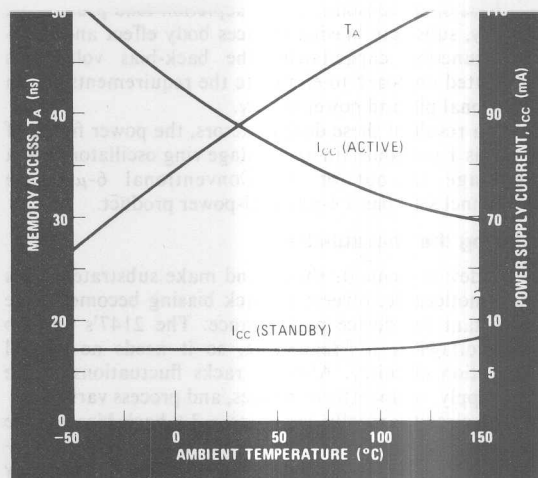
Circuit operation

Like other fully static RAMs, the 2147 can perform multiple read/write operations during a single chip-select cycle. However, with a fully static standby mode, it features two distinctly different read/write cycles: the address-access cycle and the chip-select-access cycle.

In the address-access cycle, the chip, selected previous to address change, operates just like a conventional static RAM. Address information is buffered to the decoders, the cell selected, and the data rippled through to the output. The device's power dissipation remains at a constant level throughout this read operation.

The new chip-select-access mode chip can control the device's power dissipation as well as initiating read operations. When the 2147 is deselected, it dissipates about a tenth of its active power. The access time from chip select to output stage is equivalent to that of the conventional address-access operation. Thus, there is no access-time penalty for this low-power standby feature.

The internal circuit operation during chip-select access is very simple. In such a cycle, the address inputs are valid before, or coincident with, the chip-select timing. It takes about 5 ns internally for the chip-select signal to be buffered and to activate the address buffers.



5. Standing by. While the 2147's performance—its access time—gets worse as expected at elevated temperatures, a useful feature of the part is its ability to maintain an almost constant standby current value over a wide ambient-temperature range.

By the time the address inputs have been buffered, the row- and column-select decoders have been powered up. About 30 ns from the start of the cycle the memory cell is selected and its data enters the column lines.

Since the 2147 does not contain column sense amplifiers, the memory cells drive the output buffer directly. The signal ripples through the output buffer and is presented at the output pin, typically 40 ns after chip select (Fig. 3). As expected for static circuitry, the device supply current smoothly ramps up to the active power level and stays flat until the chip is deselected.

At first glance, it would appear that the 5 ns lost in powering up the address input buffers would make chip-select access longer than address access. However, during power down all the differential nodes in the RAM are equalized as a direct result of the fully static techniques employed. By balancing the internal nodes, about 10 ns is shaved off the chip-select access time. In fact the chip-select access is typically 5 ns faster than address access (Fig. 3).

The novel standby feature was realized by utilizing the special MOS device characteristics of H-MOS. The power-down circuitry is fully static and requires no precharging or boot-strapping. Power is simply switched off in all but the essential circuit blocks. As a result, there are no power-up spikes or precharge spikes observed in the power supply current in a chip-select-access cycle. During standby, the memory array is completely deselected and the device output placed in the high-impedance state. To write into the 2147 the address inputs must be set up before the write-enable signal. Then the write operation will be completed, so long as

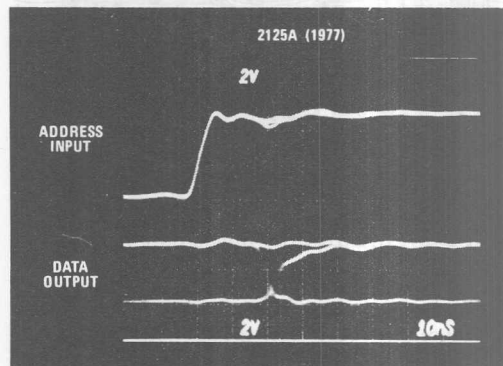
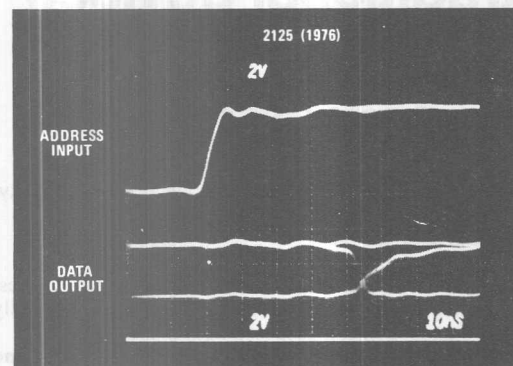
From development tool to product

The Intel 2125 random-access-memory was chosen as a test vehicle to evaluate the potential of H-MOS. During redesign in the new process, a simple 25% linear shrink was performed to take advantage of the improved density. For the same input signal, the typical access time for the data output of the H-MOS version is about half the value of the 2125 (see photographs).

The die size shrank from 18,496 square mils to 10,201 mil²—a 45% reduction. Furthermore, the performance of the 2125 improved from a typical access time of 45

nanoseconds to less than 22 ns, while the typical power dissipation at room temperature went from 325 milliwatts to 250 mW.

The development work was so successful that it was decided to make this improved high-speed 1,024-bit RAM available as the 2125A. The open-collector version, the 2115A, is also available with a guaranteed 16-milliampere output-sinking capability. Both parts are specified with a worst-case 45-ns address-access time and 393-mW power dissipation level.



the data input is valid during the write-enable pulse. A short write recovery time is required before entering another memory cycle. In the write mode, the device output is in the high-impedance state.

Write cycles can be performed in the chip-select mode, as well. Chip-select and address changes are handled normally before write enable, and the write cycle proceeds from that point as usual.

Test and reliability

The 2147 is a simple fully static RAM, so it enjoys all of the testing and reliability benefits of a fully static design. The part has little or no pattern sensitivity and can tolerate a noisy system environment. Address inputs may be skewed and rise times different; access to the RAM will take the same time as it would if the addresses came up cleanly.

A common problem plaguing static RAMs is data retention. With recent technological innovations such as H-MOS, it is possible to reduce memory-cell power dissipation to less than 10 nanowatts. This low dissipation opens the door to low-power standby features, but the 2 nanoamperes of cell current present in such a mode comes uncomfortably close to the cell-junction leakage current at elevated temperatures. The testing problem is obvious: how to guarantee data retention over extended intervals at high temperature.

Depletion-load cells, as in the 2147, can be tested under conditions that will accelerate the retention-failure time of marginal devices. The test time is reduced by using conditions that will increase cell leakage currents (primarily junction leakage) without increasing

the load-sourcing current at the same time.

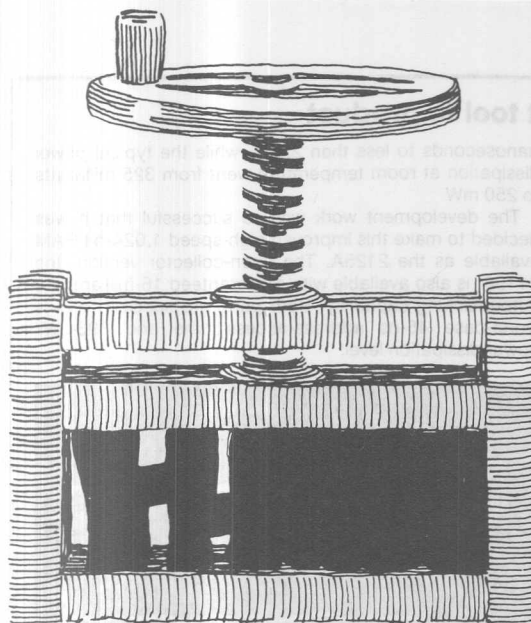
The proof is in the test results established with the family of high-performing depletion-mode static RAMs: the 2147, the 2125A, and the 2115A (see "From development tool to product" above). In fact, the reliability of both the 2125A and 2147 is equivalent to that of the highly reliable 2115. One year's accelerated life test results for the 2125A/15A predict a failure rate of 0.02% for each 1,000 hours operation at 55°C, with a 60% confidence level. Preliminary results for the 2147 indicate that it has a similar reliability probability.

Designing systems

The 2147 has been designed for both large and small memory-system applications. The single-supply device in an 18-pin, dual in-line package yields higher board densities than other dynamic or static 4-k RAM designs.

For memory systems deeper than 4 kilobits, the standby feature results in a significant power savings to the user. For example, a memory 32,768 by 9 bits deep would typically dissipate 7,560 mW, while a conventional static RAM system would dissipate 36,000 mW. The larger the memory size and the slower the cycle time, the greater this difference becomes (Fig. 4). Specifying the 2147 reduces system power and cooling requirements, as well as improving system reliability.

Like all MOS RAMs, the 2147's performance is sensitive to temperature. Both access time and active power vary widely over the military temperature range of -55°C to 150°C. However, it is significant that standby current is unaffected by temperature (Fig. 5). It remains at 8 mA over the entire military temperature range. □



H-MOS scales traditional devices to higher performance level

by Richard Pashley, Kim Kokonnen, Edward Boleky, Robert Jecmen, Samuel Liu, and William Owen
Intel Corp., Santa Clara, Calif.

□ It has almost become a law of nature, the way metal-oxide-semiconductor devices double in density or performance every year. Over the last decade, MOS chips have gone from being low-density shift registers, gates, and flip-flops operating at millisecond speeds to being entire memories, microprocessors, and dedicated systems and subsystems packing tens of thousands of electronic functions into a single device that is capable of nanosecond operation.

Fueling this astonishing progress is the tremendous versatility of metal-oxide-semiconductor technology. Starting out as a high-threshold p-channel multiple-supply circuit technique capable at best of simple calculator and serial-storage functions, MOS moved to n-channel single- and double-layer structures that use a single 5-volt power supply to perform complex computer instructions in less than 100-ns cycles and static and dynamic memory operations in less than 50 ns—all at ever lower power dissipations.

Now MOS circuit technology stands at a still higher level. For the first time it can challenge the performance of bipolar circuits, while continuing to set new records in complexity and low cost. The techniques that have proved capable of achieving this breakthrough are various but share a crucial characteristic in that they all shorten the effective channel length, or drain-source spacing, of the fundamental MOS transistor.

Two approaches are possible. One relies on a double-diffused process: a depletion-mode device with a relatively long, 5-micrometer channel under the MOS gate is integrated in series with a 1- μ m enhancement-mode channel, which is formed by the outdiffusion of boron through the self-aligned source-junction opening. The process is called D-MOS when the double-diffused structure has a planar configuration, but V-MOS when the structure has a vertical configuration, with the surface of the MOS transistor laid on the face of a V-shaped groove etched anisotropically into the silicon substrate. In either

case, the double-diffused structure requires new process technology and circuit structures that differ markedly from standard silicon-gate techniques.

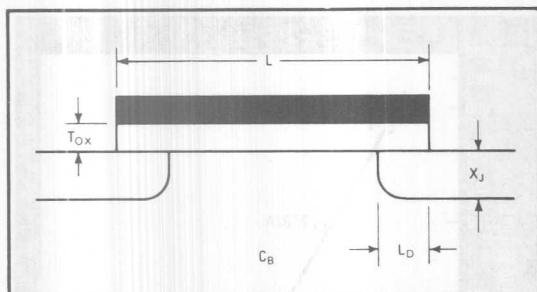
The other approach relies on scaling down the size and parameters of MOS devices directly—in other words, device-scaling conventional n-channel silicon-gate structures. This is nothing new; right from the start MOS designers knew that by trimming down the size of their devices they could achieve higher speed, higher density, and lower power dissipation.

To achieve their new high-performance process called H-MOS, Intel has chosen the direct device-scaling method for two reasons. First, it evolves directly out of standard silicon-gate processing and so requires neither new device structures nor complex circuit schemes (either requirement would make yields and fabricating costs too unpredictable to guarantee their usefulness over a wide range of semiconductor products). Second, it fits in with the trend to smaller and smaller circuit patterns, as photolithographic methods grow more refined and electron-beam wafer-fabrication techniques stand ready to take over.

Further, double-diffused structures have a limited future. They may have been appropriate two to three years ago, when the industry was unable to build channels less than 5 or 6 μ m long. But now that 4- μ m (and soon 3- and 2- μ m) channel lengths are possible, the need for new structures like D-MOS and V-MOS may simply be in the process of vanishing.

How to scale an MOS device

Figure 1 shows the cross section of a silicon-gate n-channel device, where L is the channel length, T_{ox} is the gate-oxide thickness, X_j is the junction depth, L_D is the lateral diffusion, and C_B is the substrate doping level. Now, first-order scaling theory says that the characteristics of an MOS device can be maintained and the desired operation assured if the parameters of the device are



1. Scaling down. To reduce the size of an MOS device, all physical parameters must be scaled down proportionally. If the channel length L is shortened by $1/S$ where S is the scaling factor, then the oxide thickness, T_{OX} , the lateral underdiffusion, L_D , and the junction depth, X_J , must also be scaled down by $1/S$. Meanwhile, the substrate doping constant, C_B must be increased by S .

scaled as shown in Table 1. When S is the scaling factor and the channel length L is scaled by a factor of $1/S$, then the other device dimensions—the thicknesses of the gate oxide and the lateral underdiffusion, the device width and junction depth—must also be scaled $1/S$. Moreover, to maintain adequate threshold voltage and drain-source breakdown voltage, the scaling theory also states that the substrate doping concentration must be increased by S , while the supply voltage and current decrease by $1/S$.

The effect on performance

When this is done properly, the increase in the performance of the device is dramatic, as Table 1 also shows. The parasitic capacitance, gate delay, power dissipation, and power-delay product all improve markedly. Since the parasitic capacitance goes down roughly as the junction depth decreases, it too scales by $1/S$; this means that since gate delay is roughly proportional to parasitic capacitance, it is scaled by $1/S$ as well. Moreover, since the device's power dissipation is proportional to the supply voltage and current, it scales by the still stronger factor of $1/S^2$. Finally, the bottom line of all this is the power-delay product, or figure of merit, of the MOS device; and since it is the product of the gate delay and power dissipation, it is scaled down by a very significant factor of $1/S^3$. Thus, scaling the dimensions of an MOS device improves its performance by the cube of its scaling factor.

In short, by reducing the dimensions of a circuit, the MOS designer gains enormous leverage on its density and performance—a statement that happens also to describe a recurring event in MOS history. Table 2 places the move to H-MOS in this perspective. Notice the sharp reduction in circuit parameters that occurred between 1976 and 1977 when Intel went to H-MOS from standard n-channel silicon-gate processing. By reducing the channel length from 6 to $3.5 \mu\text{m}$ and decreasing the other parameters appropriately, it was possible to quarter the speed-power product. This improvement would have been even larger had the supply voltage been scaled as required by a first-order device-scaling theory, instead of being kept at the more acceptable 5-v system

TABLE 1: MOS DEVICE SCALING

Device/circuit parameter	Scaling factor
Device dimension, T_{OX} , L , L_D , W , X_J	$1/S$
Substrate doping, C_B	S
Supply voltage, V	$1/S$
Supply current, I	$1/S$
Parasitic capacitance, WL/T_{OX}	$1/S$
Gate delay, VC/I (τ)	$1/S$
Power dissipation, VI	$1/S^2$
Power-delay product	$1/S^3$

TABLE 2: EVOLUTION OF MOS DEVICE SCALING

Device/circuit parameter	Enhancement-mode n-MOS 1972	Depletion-mode n-MOS 1976	H-MOS 1977	MOS 1980
Channel length, L (μm)	6	6	3.5	2
Lateral diffusion, L_D (μm)	1.4	1.4	0.6	0.4
Junction depth, X_J (μm)	2.0	2.0	0.8	0.8
Gate-oxide thickness, T_{OX} (\AA)	1,200	1,200	700	400
Power supply voltage, V_{CC} (V)	4–15	4–8	3–7	2–4
Shortest gate delay, τ (ns)	12–15	4	1	0.5
Gate power, P_D (mW)	1.5	1	1	0.4
Speed-power product (pJ)	18	4	1	0.2

level. However, by 1980, as channel length shrinks to $2 \mu\text{m}$ and the supply voltage to 3 v, performance will improve even more dramatically, this time by a factor of five, to become altogether 20 times better than that of 1976 MOS devices.

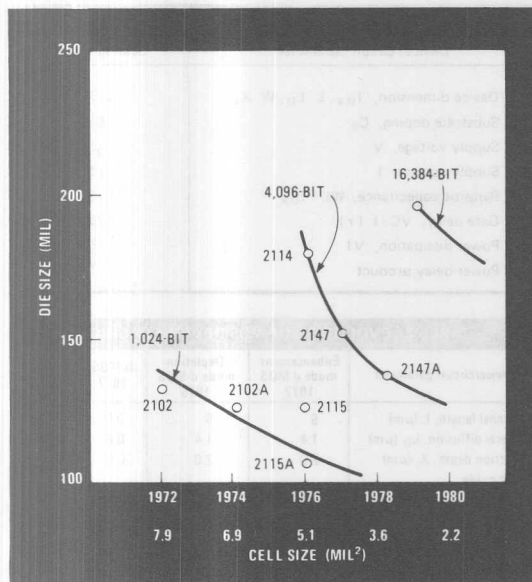
An example of what device scaling means to the user of integrated circuits is given in Figs. 2 and 3, which chart the progress made over the years in static random-access memories. In 1972, the standard static MOS RAM was the 500-ns 2102, built with a $6\text{-}\mu\text{m}$ channel length and 1,200-angstrom gate-oxide thickness. Its resulting speed-power product was 18 picojoules. It occupied a silicon chip nearly 140 mils on a side and had a cell size of almost 8 square mils.

In 1974, the 2102 was redesigned around a depletion-load n-channel technology that shrank its die area by 15% and its access time to 200 ns. To this process oxide isolation and built-in substrate bias were added in 1976, to create the 2115 static RAM that accessed the same 1,024 bits in less than 70 ns. Today, the impact of device scaling is even more apparent with H-MOS, which fits the 2115 RAM (now called the 2115A) onto a chip slightly larger than 100 mils on a side, while improving access time typically to 25 ns.

More to come

Moreover, applied to a 4,096-bit static memory design, H-MOS results in a chip a little larger than the original 2102, yet pushes access times typically below 50 ns. Finally, as the MOS process evolves and scaling continues, a 16,384-bit fully static RAM will fit on a chip no larger than 200 mils on a side and offer system designers access times in the 50-ns range.

The high speed and high density of H-MOS are achieved through five major improvements in MOS technology, four of which are directly related to device



2. Better and better. Thanks to the vigorous development of n-channel silicon-gate MOS technology, static RAMs continually improve in density. In comparison with earlier processes, today's H-MOS increases device packing density by a factor of 4.

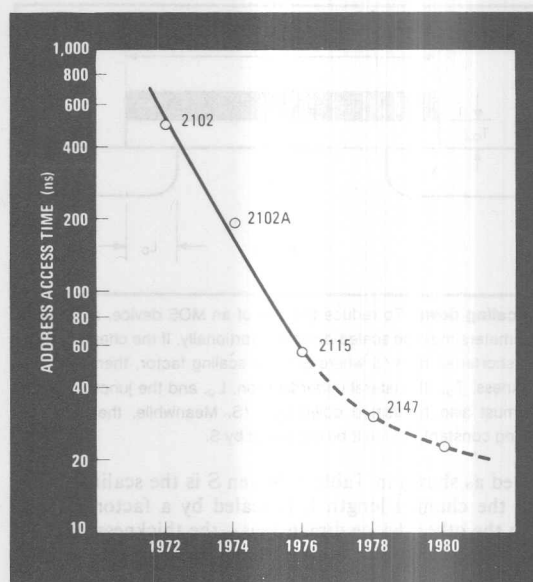
scaling. A high-resistivity substrate is used, while device-scaling theory is applied to gate-oxide thickness, junction depth, gate length, and threshold-modifying ion implants.

The high-resistivity substrate made of 50-ohm-cm, p-type material is used to lower junction capacitance, reduce the substrate body effects that degrade performance, and increase the device's effective carrier mobility. All three factors result in faster, lower-power devices.

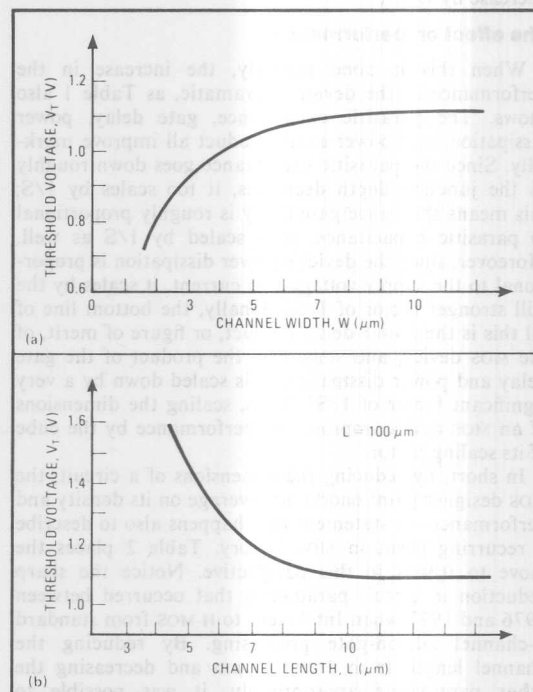
Scaling the H-MOS gate oxide down to 700 angstroms increases device gains and punch-through voltages and reduces body and short-channel effects, thereby increasing performance and reliability. The junction depth is scaled to approximately $0.75\text{ }\mu\text{m}$ by using slow-diffusing arsenic as the source-drain dopant. The shallowness of the junctions increases both speed, by reducing peripheral junction capacitance and gate-drain Miller capacitance, and density, by allowing smaller diffusion-to-diffusion spacing.

Scaling principles are also applied to the polysilicon-gate electrodes, which form the self-aligned source and drain diffusion regions. The narrow $3.5\text{-}\mu\text{m}$ polysilicon gates of H-MOS increase the device gain and still further increase circuit speed and density. Narrow gates, however, come at the expense of more severe photolithographic and etch control requirements, which are needed to avoid a wide variation in the electrical channel lengths of the device.

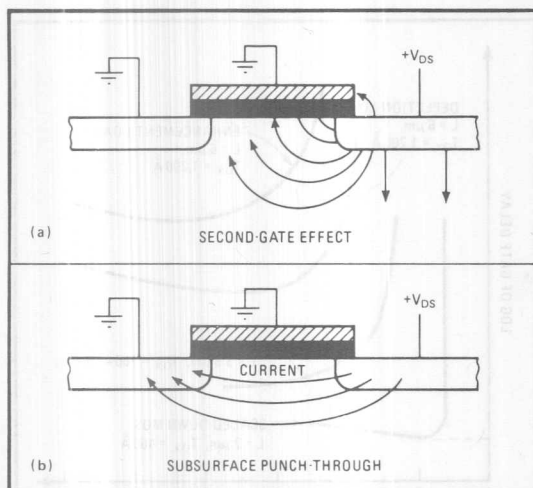
Finally, H-MOS threshold-voltage stability is maintained for both enhancement-mode and depletion-mode devices by using an ion-implanted channel region in conjunction with the high-resistivity substrate. This implant procedure controls threshold voltage with great



3. Faster, too. Process improvements and device scaling, as embodied in H-MOS, are also making MOS RAMs faster. In 1972, a typical 2102 1,024-bit static RAM had an access time of 600 ns; today's 2147 4,096-bit parts can be accessed typically in 45 ns.



4. Maintaining that threshold. Decreasing channel length and width in small devices has a strong effect on threshold voltage. When the channel length goes below about 5 micrometers, V_t begins to decline (a); while for widths below $7\text{ }\mu\text{m}$, V_t begins to climb (b).



5. Second-order problems. Small devices are vulnerable to two second-order effects. One is the second-gate effect (a), where the electric field lines emanating from the drain junction end up on the oxide-silicon interface. The other is punch-through (b), which can be relieved by careful choice of the substrate impurity profile through ion implantation combined with a thin gate oxide.

precision and allows the MOS threshold voltage to be optimized independently of the substrate doping.

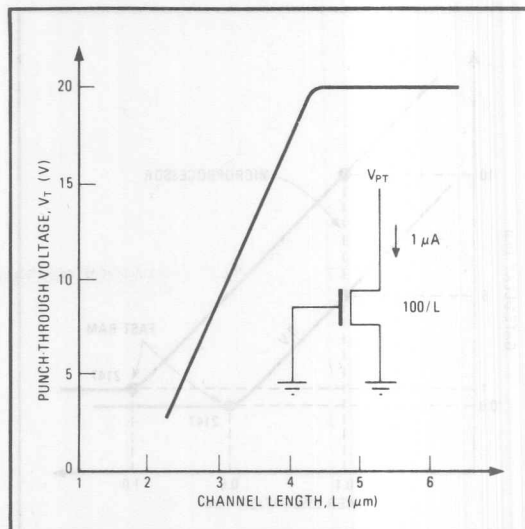
Happily, it proved possible to make all of these H-MOS technology advances within a relatively short time (about one year) without affecting the ability to manufacture devices at reasonable costs. H-MOS also is flexible enough to be applied over a broad range of circuit designs while maintaining its inherent high speed, small size, and low power. Unlike V-MOS and integrated injection logic, which require new circuit techniques to make them applicable to dynamic-memory and large-scale-integrated logic designs, H-MOS can be directly applied across the entire product spectrum.

Already the process has resulted in a family of static RAMs (the 1-k 2115A and 4-k 2147), which offer the industry the best speed-power performance of any memory. Moreover, work is under way on the application of H-MOS to a high-performance, 16-bit microprocessor family, a large variety of complex peripheral chips, 16-k and 65-k dynamic RAMs, high-density ready-only memories, and erasable programmable ROMs.

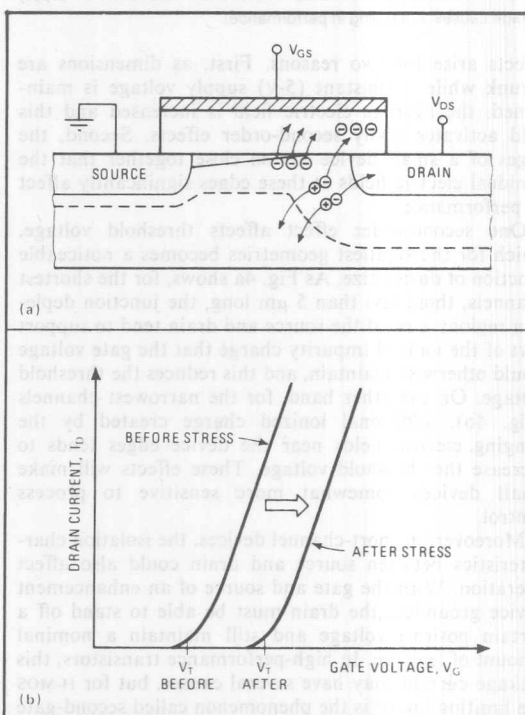
Its ability to be upgraded is a final and very important feature of H-MOS. Indeed, H-MOS is only the first step in that direction. As advances in photolithography occur, direct scaling can be applied to improve speed-power product and density even further.

Beyond first-order theory

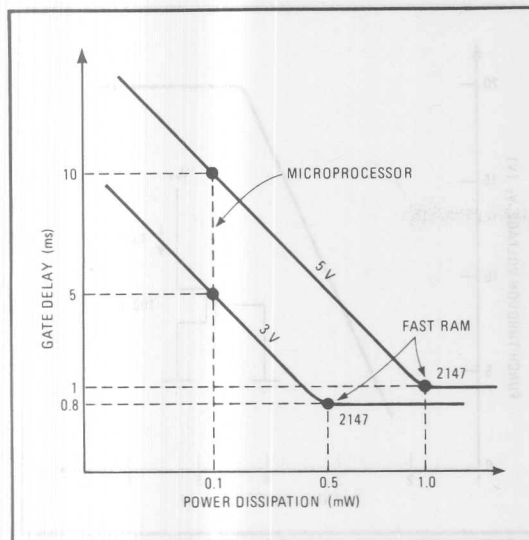
As devices are shrunk, scaling theory says ideally they should maintain the same qualitative characteristics. But in reality, second-order phenomena become quite significant. Some of these phenomena affect the circuit design, while others relate to reliability, but all have to be considered and understood to assure that H-MOS is a useful and safe process. Basically, all of the second-order



6. Guaranteeing punch-through. In short-channel MOS devices, the punch-through voltage falls to levels that could cause high leakage and circuit problems. The answer is to keep the channel length in the $4\text{-}\mu\text{m}$ region and to see that the gate oxide is thin.



7. Effect of trapped electrons. So much charge can be trapped in the gate oxide of short-channel devices (a) that it may cause a permanent shift in the threshold voltage (b). This shift could cause a reliability problem in these devices, but it can be minimized by very careful oxide processing and by reducing the supply voltage.

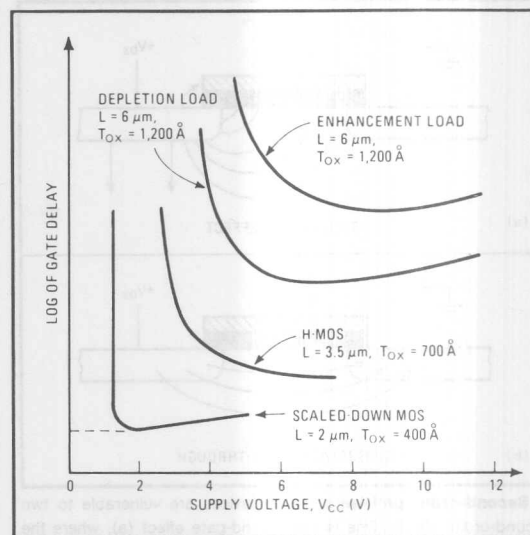


8. A change in supply voltage. Lowering the supply voltage from 5 to 3 V significantly enhances the speed and power dissipation of MOS circuits, especially scaled-down devices. The effect is most noticeable for microprocessors, where a 2-V reduction in supply voltage causes a doubling in performance.

effects arise for two reasons. First, as dimensions are shrunk while a constant (5-v) supply voltage is maintained, the average electric field is increased and this field activates many second-order effects. Second, the edges of a small device are so close together that the nonideal electric fields at these edges significantly affect its performance.

One second-order effect affects threshold voltage, which for the smallest geometries becomes a noticeable function of device size. As Fig. 4a shows, for the shortest channels, those less than 5 μm long, the junction depletion regions around the source and drain tend to support part of the ionized impurity charge that the gate voltage would otherwise maintain, and this reduces the threshold voltage. On the other hand, for the narrowest channels (Fig. 4b), additional ionized charge created by the fringing electric fields near the device edges tends to increase the threshold voltage. These effects will make small devices somewhat more sensitive to process control.

Moreover, in short-channel devices, the isolation characteristics between source and drain could also affect operation. With the gate and source of an enhancement device grounded, the drain must be able to stand off a certain positive voltage and still maintain a nominal amount of leakage. In high-performance transistors, this leakage current may have several causes, but for H-MOS the limiting factor is the phenomenon called second-gate punch-through (Fig. 5). In it, the electric field lines emanating from the drain junction terminate at the oxide-silicon interface of the channel. There the drain acts as an unwanted second gate and inverts the channel from the back, making the device more sensitive to punch-through effects.



9. Benefiting performance. As MOS technology becomes better and devices smaller, the need for lower supply voltages becomes more urgent. For 2-micrometer channel lengths, a 3-V supply yields the best gate-delay performance and process reliability.

The problem for short-channel devices is that the punch-through voltage arising from this effect is a linear function of the channel length (Fig. 6). The shorter the channel, the lower the voltage causing punch-through and therefore the more susceptible is the device to leakage. In H-MOS however, this is overcome by maintaining a long enough channel length and reducing the oxide thickness, since a thinner oxide prevents unwanted inversions by capacitatively coupling the surface potential more tightly to the grounded gate electrode. A second punch-through effect occurs, as shown in Fig. 4b, when the electric field from the drain reaches through to the source and forward-biases the junction, causing current to flow—it is similar to that in a bipolar transistor; but again this punch-through voltage, which is proportional to L^2 , is a limiting factor only for devices smaller than those that are being used at present in H-MOS designs.

Impact ionization

Another source of leakage is impact ionization, the effects of which are illustrated in Fig. 7a. At a very large drain voltage of around 20 V, the junctions avalanche for all channel lengths greater than about 4 μm . But even at the significantly lower (5-V) drain voltages of H-MOS, weak impact ionization can occur when current is flowing through the device channel. Activated by the high electric fields, impact ionization creates a population of electrons and holes with energies much higher than the normal channel electrons. The holes flow into the substrate and place a small load on the back-bias supply. Some of the electrons have enough energy to be injected into the gate oxide, as shown, where they can cause a gate current or be trapped. These trapped electrons cause a shift in the threshold voltage (Fig. 7b)—a

three metal-oxide-semiconductor approaches to high performance are H-MOS, V-grooved MOS, and silicon on sapphire. As the accompanying table shows, the current versions of H-MOS and V-MOS both yield a speed-power product of about 1 picojoule.

V-MOS, in principle, has a slightly better packing density but pays for this compactness with a more complex process. Also, V-MOS yields an asymmetric device that must be used in one direction only, so that large-scale-integrated logic configurations are much more difficult to achieve than with H-MOS.

SOS, on the other hand, has the best speed-power product. But it requires a substrate five to seven times more costly and seems justified only for microprocessor applications, which do not require operation at the high-speed end of the speed-power curve.

The main advantage H-MOS has over V-MOS today is the fact that the scaling-down process moves it directly to higher performance and greater density at lower cost. The performance for 1980 scaled-down MOS (2-micrometer channels) is shown—it is about five times better than today's technology.

THREE MOS TECHNOLOGIES COMPARED

Parameter	H-MOS 1977	Scaled- down H-MOS 1980+	V-MOS 1977	SOS 1977
Layout density (gates/mm ²)	170	200	~220	150
Speed power product (pJ)	1	0.2	~1	0.2
Gate delay (ns)	1	0.4	~1	0.5
Number of thin films	2	2	3	3
Number of implants	3	3	3	2

shift that could pose a reliability problem with channels less than 4 μm long.

Finally, there is the increase in interconnect capacitances induced by fringing fields. This parasitic effect occurs for some of the same reasons as the increase in threshold voltage associated with narrow channels—a narrow metal line over the large silicon ground plane has a larger effective area and therefore a larger parasitic capacitance from the fringing fields near its edges.

Happily, the only potential reliability problem brought out by the second-order theory—the trapping of injected electrons in the gate oxide—turns out not to affect H-MOS in its present form. True, trapped electrons tend to increase the threshold voltage, and an increase in threshold voltage could degrade circuit speed or totally stop it from functioning. But accelerated stress tests on H-MOS memory circuits reveal no signs of degradation [*Electronics*, Aug. 4, p. 103]. In fact, additional measurements on individual transistors, plus a physical model for electron injection, show that H-MOS devices will have a total threshold shift of less than 0.1 v after 10 years of continuous stress at worst-case conditions (at 0°C and a V_{DS} of 5.5 v). Careful processing of the gate oxide partly

devices are subjected to less stress than are the oxides of today's 12-v dynamic RAMs, since 5 v across an oxide 700 angstroms thick is less of an electric field than 12 v across the standard 1,000-angstrom oxides.

While the scaling down of devices as used in H-MOS for boosting MOS performance has a bright future, one condition must be met if its full potential is to be realized. That condition is a reduction in power supply voltage. Table 2 shows that if the technique is to work at all, the supply voltages for 1980 2- μm devices must be scaled down to the 2-to-4-v range. Since all supply voltages are now maintained at least at the 5-v TTL level, this lower supply-voltage requirement for future MOS devices must be accepted by integrated-circuit users.

There is, of course, an alternative for users who simply refuse to accept low system supply voltages. They could use converters to translate between the lower chip-voltage levels and the higher TTL input/output levels, or they could use two power supplies, one providing the chip's internal circuitry with 2 to 4 v and the second supplying 5 v to their I/O circuits. But either procedure is makeshift at best.

Key to the future: lower supply voltage

The fact is, a lower power supply voltage significantly increases the reliability of small-pattern devices, while at the same time increasing their performance remarkably. Reliability goes up because lower supply voltages entail greater tolerance to lower punch-through voltage and at the same time yield weaker electric fields in the channel region. This second effect reduces the risk that charge will be trapped in the gate oxide and alters the long-term stability of the device.

The increase in performance is even more striking. For RAMs and microprocessors (Fig. 8), the impact on the speed-power product would alone make it worth while going from a 5-v to a 3-v supply voltage. For RAMs, which operate at the saturation point of the speed-power curve, the lower power-supply voltage reduces power dissipation by about 60%, while maintaining the speed at the same high value. This reduced power dissipation becomes extremely important as the chip density goes up—65,536 bits and 262,144 bits—since it is generally agreed that for reliable operation power dissipation per package must be kept below a watt.

As for microprocessors, since they operate in a region that is well removed from the saturation point of the speed-power curve, they can take full advantage of a significant speed increase for a given power dissipation. A 3-v microprocessor chip, for example, will operate at twice the speed of a 5-v device.

How the supply voltage affects the various MOS processes that have evolved over the years is shown in Fig. 9, and again the desirability of lower voltages becomes evident. Indeed, for the 1980 scaled version of MOS devices—channel lengths of 2 μm and oxide thicknesses of 400 angstroms—a power supply of 2 to 4 v will give half the gate delay of today's H-MOS process operating at 5 v. □

Single-supply, 16-k dynamic RAM is ready for denser systems

While 16-k dynamic RAMs are firmly established in memory system designs, RAM technology continues to be improved in response to user demands for a wider range of choices among devices. It's not simply a matter of making RAMs faster and easier to use. Users also want RAMs that are compatible with both current devices and the coming, denser units. The first dynamic RAM to satisfy all these requirements is the Intel 2118.

Organized as 16 kwords \times 1 bit, the 16-pin 2118 is designed to operate in systems requiring 100-ns access. It is also the first 16-k \times 1 RAM to operate with a single +5-V supply and to offer very low maximum levels of operating (130 mW) and standby power (15 mW).

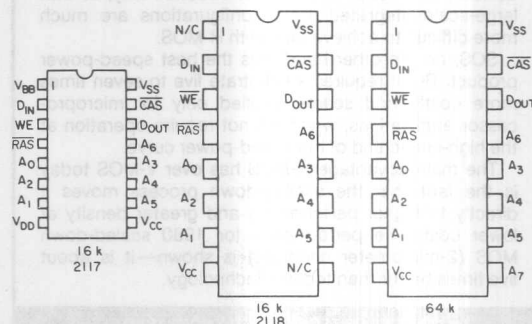
Except for its supply, the Intel 2118 is functionally compatible with existing 16-k devices, such as the Intel 2117. Not only that, the 2118 will also be voltage and pinout-compatible with future 64-kbit RAMs. What that means is that high-performance, 5-V-only memory systems can be designed now and have their density increased simply by plugging in future 64-k dynamic RAMs.

The compatibility with both 16-k and 64-k RAMs can be seen in the pinout and package designs for the 2118 (see Fig. 1). Indeed, the only functional difference between the 2117 and the 2118 is that the 2118 requires just the one 5-V supply. For the devices in Fig. 1, 128 refresh cycles are required every 2 ms. With such compatibility, performance can be upgraded with minimal wiring changes on the control and memory cards.

One supply

Most NMOS dynamic RAMs, including the current 16-k's, use -5-V substrate bias and +12 V of drain voltage. The smaller size of the MOS transistors lead to density, power, and performance improvements, but require a lower positive supply to avoid source-drain punch-through.

The 2118's positive supply has been lowered to +5 V, and the negative-voltage substrate bias is internally generated—its operation is both automatic and trans-



2118	
A ₀ -A ₆	ADDRESS INPUTS
CAS	COLUMN ADDRESS STROBE
D _{IN}	DATA IN
D _{OUT}	DATA OUT
WE	WRITE ENABLE
RAS	ROW ADDRESS STROBE
V _{CC}	POWER (+5V)
V _{SS}	GROUND

1. **Pinout compatibility** allows 2118 RAMs to replace present-generation 2117 16-k dynamic RAMs and also lets forthcoming 64-k RAMs replace both without significant changes.

parent. And without an externally supplied substrate bias, the 2118 can fit into newer high-performance microprocessor systems without additional power supplies. In mainframe memory systems also, the single supply simplifies the lay out of storage boards, while cutting power-supply costs.

The 2118's sub-100-ns access, is much faster than current 16-k RAMs (see Fig. 2). But the 2118 doesn't dissipate nearly as much power as current 16-k dynamic RAMs, most of which dissipate as much as 460 mW. The 2118 dissipates no more than 130 mW, and just 16 mW during standby, which reduces power-distribution costs and cooling requirements. Moreover, the 2118 draws very small transient currents through its +5-V input (see Fig. 3).

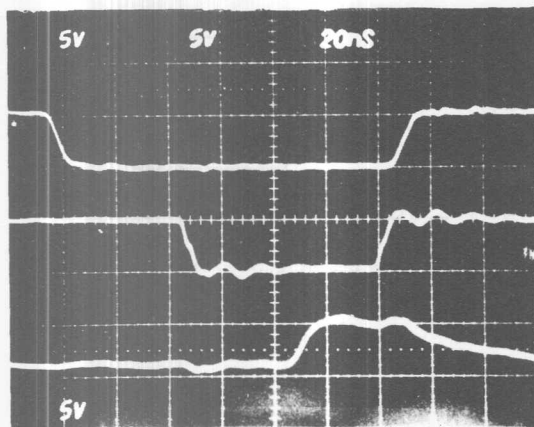
Design with the 2118

To see how the 2118 fits into the design of a high performance memory system, take a 64-kword memory with 16 bits per word and 130 ns access time. Bear in mind, first of all, that the timing conditions for

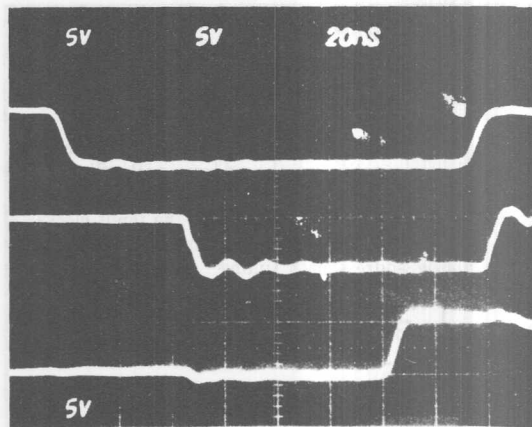
Edward Metzler, Product Manager, and James Oliphant, Marketing Manager, Intel Corp., 3065 Bowers Ave., Santa Clara, CA 95051.

REPRINTED BY PERMISSION

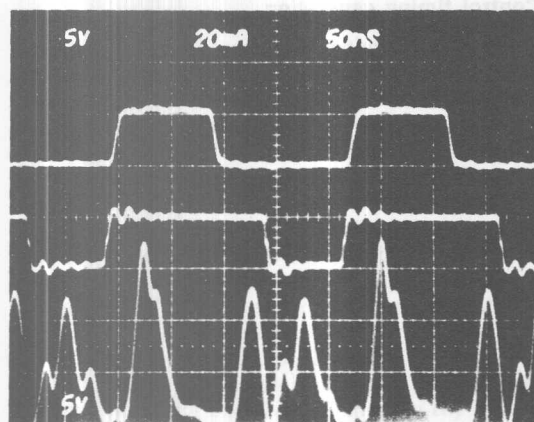
ELECTRONIC DESIGN 19, September 13, 1978



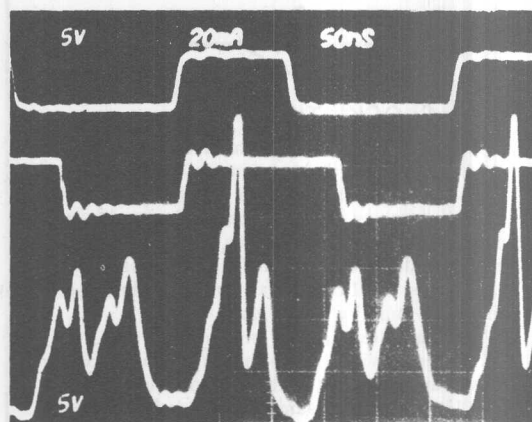
2. Access time comparison for 2118, left, and 2117 right. The 2118 shows row address strobes, top, column-address strobes,



center, and output data waveforms, bottom. Note that the 2118 has an access time less than 100 ns.



3. Current waveform comparisons (bottom) show I_{CC} for 2118, left, and I_{DD} for 2117, right. Top waveform is row



address strobe and middle waveform is column address strobe. Note the 2118's lower transients.

multiplexing the 2118 must be satisfied while using relatively slow TTL interface circuits.

In any high-speed multiplexed dynamic RAM memory, the timing problems primarily stem from the delays from a memory start signal to the appropriate device clocks. In a 64-kword memory system four time periods in all (Fig. 4) must be minimized:

- t_1 , the delay from memory start to the latest occurrence of control clock RAS at the 2118 input.
- t_2 , the address-hold time between the latest occurrence of RAS and the earliest occurrence of address changes in column addresses.
- t_3 , the time between row-address and column-address multiplexing (skew).
- t_4 , the time between the latest occurrence of row addresses multiplexed to column address and the earliest occurrence of control clock CAS.

So to take full advantage of the 2118, you must configure the interfaces to minimize not only absolute

delay through peripheral circuits but also skew through the logic circuits/drivers (for more on skew, see box). To accomplish this:

- Select all logic gate types for minimum delay and skew for the function desired. (In some cases, this means that it may be more desirable to use a simpler TTL logic gate than a more complex TTL gate that has more skew.)
- Minimize the output loading on these gates.
- Minimize skew by placing parallel gates that lie in the critical timing path in the same IC package. For high-performance control logic that includes all these factors, see Fig. 5.

The control logic is designed to drive the memory board. Note that in this configuration, the control clocks RAS and CAS for a given side and row on the board are driven from the same IC package to minimize skew (see Fig. 5). Likewise, the seven addresses required for all eight 16-k RAM devices in a particular

dynamic-RAM memory systems, skew is important when considering the timing between the control clock's logic path and the address path. Since in dynamic RAM systems, addresses must arrive at or before a specified time relative to the input clocks, the *minimum* clock timing must not exceed the *maximum* address path timing:



ADDRESS TIME



CLOCK TIME

The logic gates used in the address-timing path presumably have their maximum delays. The logic gates in the clock timing path presumably have their minimum delays. Since addresses must arrive at the memory device at or before the clocks, the clock may have to be delayed to allow for the addresses to become valid. This "artificial" delay directly reduces access time and must be minimized.

There are two ways to minimize skew:

1. Select devices whose minimum and maximum delays are as close together as possible.
2. Use gates in the same IC package for both paths. For example, if logic gates labeled A in the figure were in the same IC package, one can have a maximum delay, and the others will also be very close to maximum.

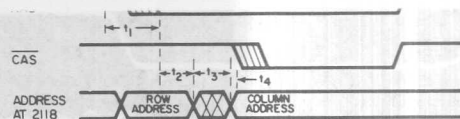
row of the card come from drivers in the same IC package (see Fig. 6).

For the address-multiplexing portion of the control logic, the latches, multiplexers and drivers have also been chosen for their minimum delay and skew characteristics.

As you can see in Fig. 6, system addresses MA₀ to MA₁₅ for the 64-kword card are brought onto the card and latched by 74S257 gates. The two 74S158s do address multiplexing between the low-order row addresses and the high-column addresses.

Refresh addresses (generated by counters) are OR'ed to the latch outputs of the low-order system addresses as shown. Using a two-input multiplexer instead of a four-input will minimize skew (save about 4 ns) when switching from row to column addresses.

Finally, the addresses are buffered by 74S04's driving the memory array. Capacitive loading is



4. **Memory timing design** requires attention to four time periods which all must be minimized. Skew must also be minimized.

minimized on these addresses by having each driver drive a moderate amount of memory devices—in this case, just 16.

Control timing generation

To generate timing for the row and column address strobe clocks and for row and column address multiplexing, use precision delay lines as shown in Fig. 5. The taps on these delay lines are at 5 ± 1 -ns increments, so you can achieve maximum flexibility in timing by choosing taps. Timing stability is excellent, since the delays remain within ± 1 ns of nominal, with respect to the delay line input.

During a read or write cycle, the appropriate RAS signal is activated on the selected row of devices on the storage card. This row is selected by the decoder (Fig. 6) from system addresses MA₇ and MA₁₅. Moreover, you'll be able to bring addresses MA₁₆ and MA₁₇ onto the card to allow for upgrading to higher-density (64-k) memory devices; a jumper is provided to help this upgrade.

Again, the RAS and CAS clock drivers for a given row of memory devices are contained in the same IC package, which saves about 10 ns of skew in these paths.

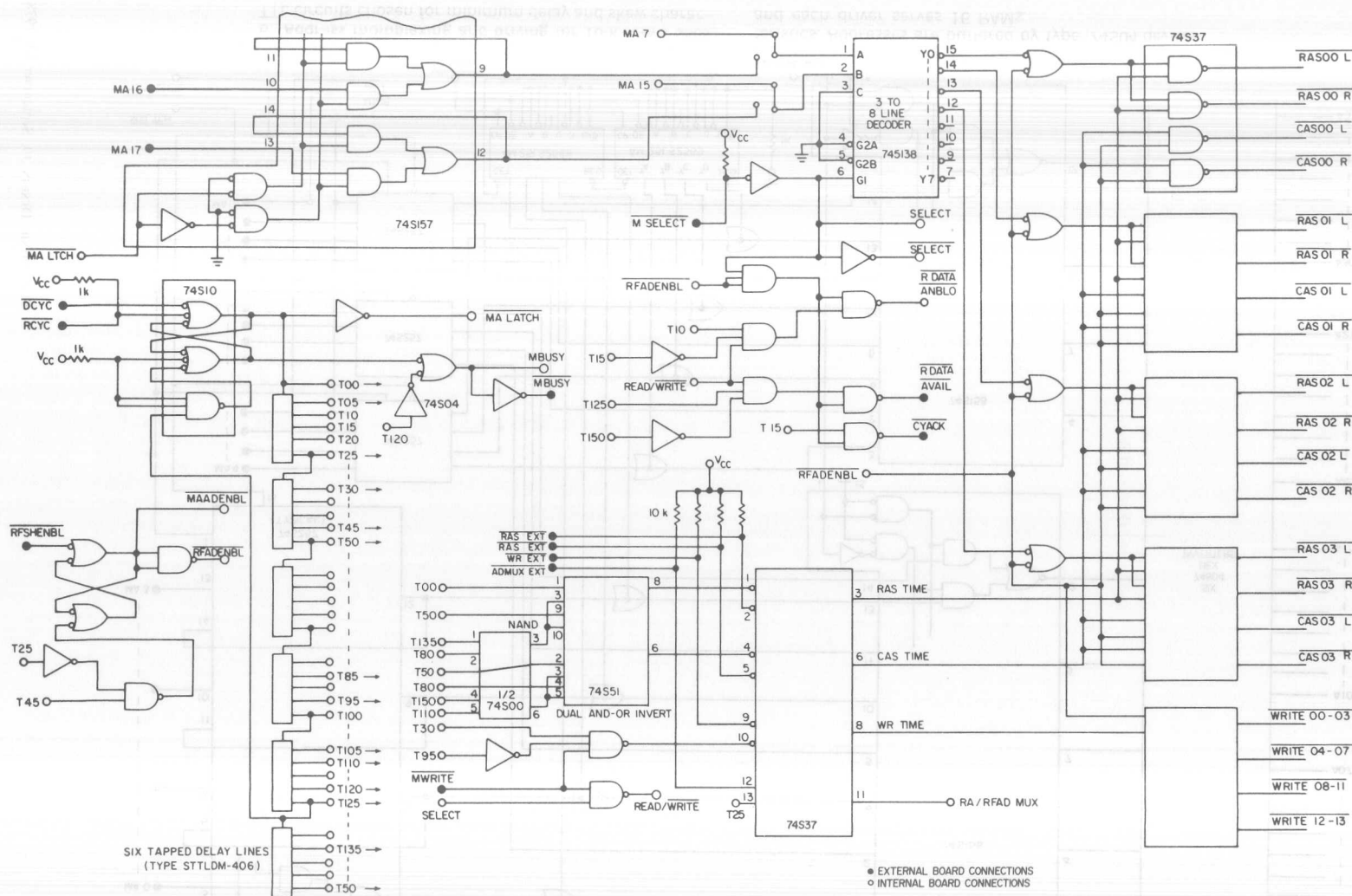
In this control logic, refresh is synchronized with a system clock. This type refresh eliminates delays associated with refresh arbitration (between a read/write cycle and a refresh cycle) that could be as high as 40 to 60 ns.

Timing calculations

With the control logic designed, we calculate worst case system delays. There are two ways to perform such calculations:

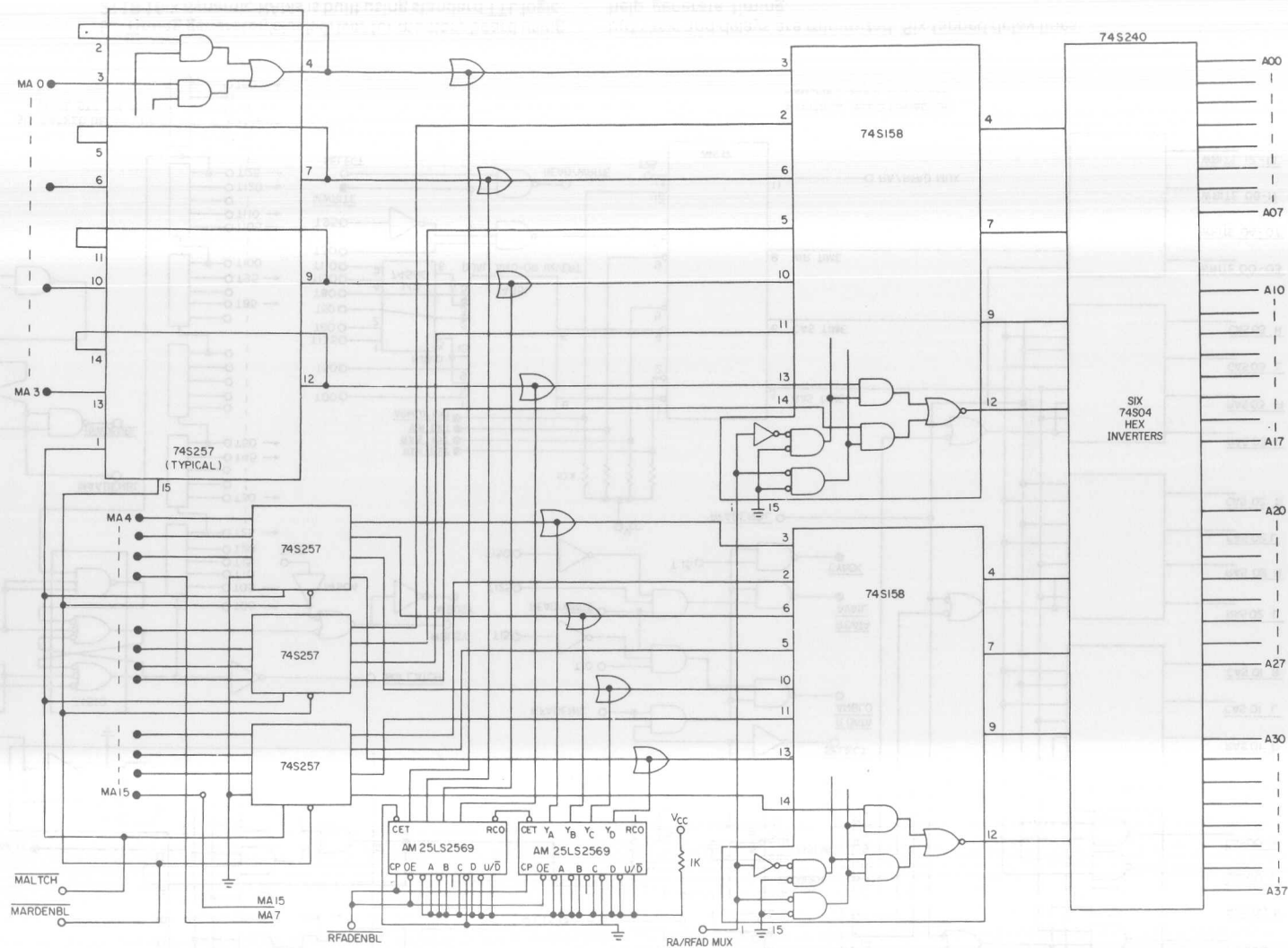
1. A true worst-case analysis, using specified maximum and minimum delays for peripheral circuits plus all delays due to capacitive loading from driven device input capacitance and PC-board etched conductor patterns.

2. A statistical worst-case analysis, which assumes that all devices can't be in their worst-case condition at the same time. Since the statistical approach can



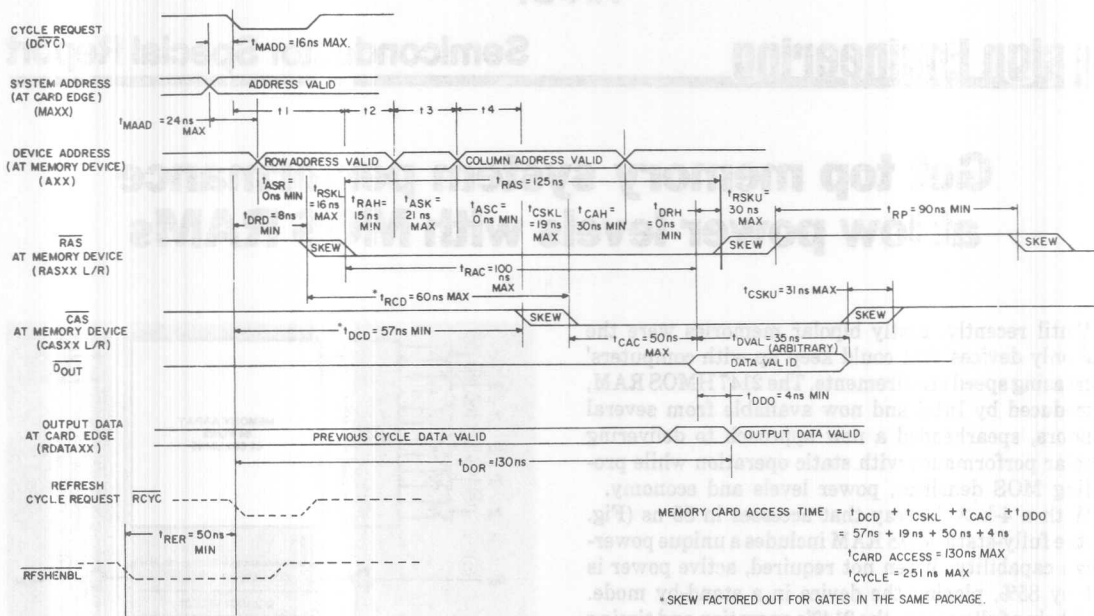
5. Timing generator/clock driver for memory board using 2118 16-k dynamic RAMs is built using standard TTL logic.

but skew and delays are minimized. Six tapped delay lines help generate timing.



6. Address multiplexing and driving for 16-k RAMs uses TTL circuits chosen for minimum delay and skew characteristics.

Addresses are buffered by type 74S04 devices and each driver serves 16 RAMs.



7. Complete memory-system timing chart shows how worst-case card access time of 130 ns can be achieved

using 2118 16-k dynamic RAMs. Timing analysis is based on TTL data-book entries.

Guidelines for timing analysis

To make a timing analysis in a logic design for worst-case conditions, assume that the TTL devices have certain characteristics. To get the fast times shown in Fig. 8, use the following guidelines:

1. All propagation delays are taken from industry TTL data books:

Max = Data book entry
Typ = Data book entry
Min = 1/2 of data book typical value, which is generally considered good practice.

2. Device-to-device skew (same package) - 0.5 ns max for Schottky TTL; 2 ns max for 74S240 buffer driver.

3. The STTLDM-406 is a special 25-ns delay line

with active outputs (available from Engineered Components Co., San Luis Obispo, CA) whose propagation delay is 5 ns \pm 1 ns per tap. (Within a line, the tolerance is not cumulative; for example, the delay from the input to the third tap is 15 ns \pm 1 ns.)

4. Capacitive loads add 0.05 ns/pF to the propagation delays specified in the data books. Schottky-TTL input capacitance is 3 pF. Printed-circuit board traces are 2 pF/in.

5. PC-board etch adds no skew to array address/control timing signals. The etch adds 4 ns to over-all access time.

6. Timing components are immediately adjacent to each other. The etch delays in the delay-line timing chain are negligible.

be justified only in large systems with hundreds or even thousands of components, the timing calculations used here are based on true worst-case analysis.

The simplified timing diagram for the control logic shown in Fig. 7 is similar to that shown in Fig. 4, but presents the specific timing conditions of the control logic in greater detail.

The timing conditions shown in Fig. 7 assume a loading effect on the TTL drivers of 0.05 ns/pF. The maximum capacitive load specified in TTL data books for high-current MSI devices is 50 pF. The RAS/CAS

drivers each drive 68 pF, which represents the effects of clock input capacitance of the 2118 and 2 pF/in. of printed circuit etch lines on the board.

Fig. 7 shows that using a 100-ns 2118, you can get a true worst-case card access of 130 ns, with the multiplexing and driving overhead minimized. In fact, you can cut the access times to 115 ns if you adjust the taps on the delay lines to "personalize" the card for a particular combination of peripheral devices. In very high-performance systems, that's the only way to get minimum access time.■

Get top memory system performance at low power levels with MOS RAMs

Until recently, costly bipolar memories were the only devices that could keep up with computers' increasing speed requirements. The 2147 HMOS RAM, introduced by Intel and now available from several vendors, spearheaded a new approach to delivering bipolar performance with static operation while providing MOS densities, power levels and economy.

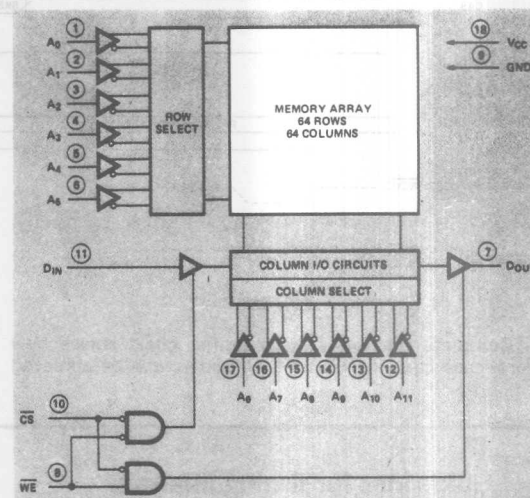
With a 4-k \times 1 array that accesses in 55 ns (Fig. 1), the fully-static MOS RAM includes a unique power-down capability. When not required, active power is cut by 85%, placing the device in a stand-by mode. And being fully static, the 2147's operation and timing requirements are straightforward.

Large numbers of these RAMs are going into cache and control store memory sections of large hierarchical computer systems. With the high-speed RAMs, the processor can operate at its fastest cycle time and process data much faster. And applications don't end there. Since it's a comparatively low-cost NMOS product, even main computer memories are starting to use these RAMs.

Approaching the Ideal

Judging from the multitude of alternate sources, the 2147 approaches an ideal memory—it is fast, fully static, operates from a single +5-V \pm 10% power supply and has an automatic low standby power mode. Actually, there are four versions available—the 2147, a 70-ns access time model; the 2147L, a 70-ns low-power option version, the 2147-3, a full 55-ns device; and a full MIL temp version, the M2147, which has an 85-ns access time. Housed in the industry-standard 18-pin DIP pinout, the RAM design uses a conventional six-transistor cell.

The 2147's high performance comes by way of the HMOS technology developed by Intel. A scaled version of previous NMOS technologies, HMOS uses 3.5-micron channels, 700-Å oxides and 1-micron junction depths. Performance is further enhanced by the reduction of junction capacitances through the use of an on-chip back-bias generator that typically provides a



1. Offering a blazing access time of 55 ns, Intel's 2147 static, 4-k HMOS RAM cuts power consumption with its automatic power-down operating mode.

–3-V back-bias voltage.

Because the 2147 is fully static, basic device operation is particularly simple. Data are simply accessed from either the Chip Select or Address Valid signals, whichever comes last—as the Read Cycle waveforms show in Fig. 2. Clocks, address setup, address hold, and address multiplexing are not required. Therefore, performance degradations due to system skews are minimal.

Since the \overline{CS} input is not a clock and does not have to be cycled, multiple read or write operations can be performed during a single select period. No time is lost between operations for a pre-charge requirement, which allows the 2147 to be cycled at its access time for improved performance over clocked, or “edge-activated,” static RAMs.

Historically, fully static RAMs have meant constant power dissipation at high active levels. The first fully static RAM to break with this tradition, the 2147 combines an innovative design approach with the benefits of HMOS to achieve low power—without the need for a clock.

Kirk F. MacKenzie, Static RAM Marketing and Applications Manager, Intel, 3585 S.W. 198th, Aloha, OR 97005.

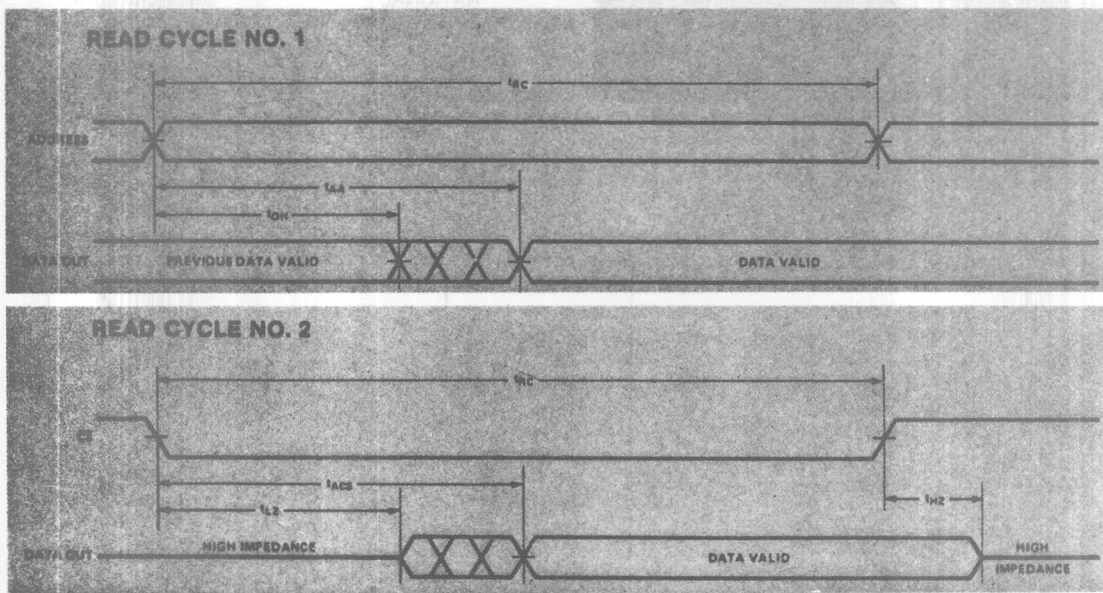
Reprinted with permission from *Electronic Design* 7, March 29, 1979

ELECTRONIC DESIGN 7, March 29, 1979

the internal peripheral circuitry are turned on (see scope trace in Fig. 3). I_{SB} remains stable over voltage and temperature variations.

The 2147's automatic power-down feature saves in two ways. It reduces power requirements as the duty

during selection, the array is in a high impedance state. To keep the supply within tolerances during these transitions, localized high frequency decoupling is required. Adding one 0.1- μ F ceramic capacitor to the board for every other RAM, and one 22- μ F bulk



2. Data are accessed from either the Chip-select or Address-Valid signals, depending upon which comes last.

cycle decreases (Fig. 4a) because the memory spends more time in the deselected, low-standby-power state. As the duty cycle approaches zero, average power dissipation approaches the standby level. Also it saves power in larger memories where only a fraction of the total memory is active at any time—typically 4 kwords. Additional memory beyond the active block is added at standby power levels (see Fig. 4b).

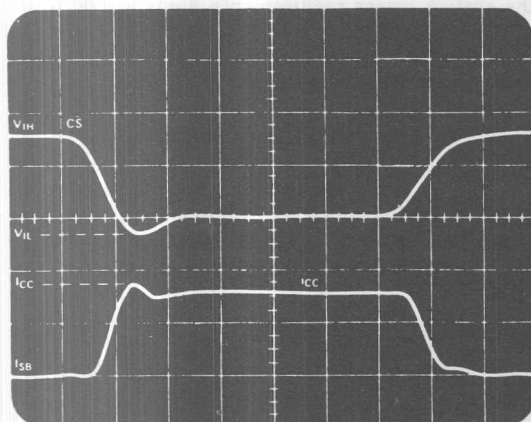
Power savings in a system

To get a good idea of the power savings possible, examine a typical system that uses the 70-ns 2147L, which has a 100-mA typical active current and a 7-mA typical standby current. For a 64-k \times 16 memory (256 RAMs), the first 4 k of memory requires 0.86 A, typical, assuming a 50% duty cycle. This is calculated by multiplying 16 devices by $(50\% \times 100 \text{ mA} + 50\% \times 7 \text{ mA})$. The remaining 60-k \times 16 memory requires only another 1.68 A, typical $(240 \text{ devices} \times 7 \text{ mA})$. The total system requirement is 2.8 A at 5 V, or 12.4 W. Without the auto power-down feature of the 2147,

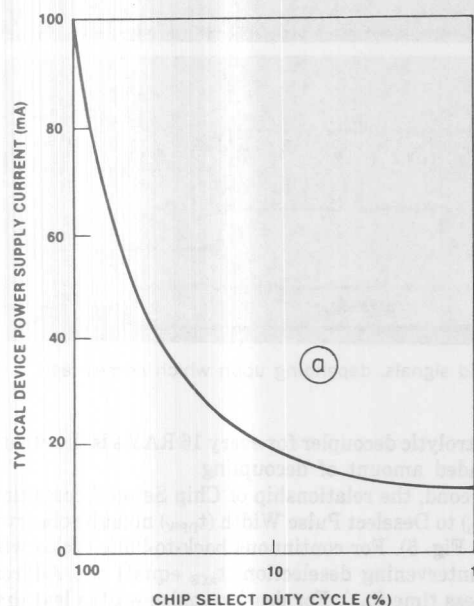
electrolytic decoupler for every 16 RAMs is the recommended amount of decoupling.

Second, the relationship of Chip Select Access time (t_{ACS}) to Deselect Pulse Width (t_{DPW}) must be observed (see Fig. 5). For continuous back-to-back cycles with no intervening deselection, t_{ACS} equals the Address Access time (t_{AA}). For deselect pulse widths less than a cycle time, t_{ACS} typically increases 5 ns because of the time lost in repowering the array. Even deselect pulse widths as narrow as a few nanoseconds are affected by this.

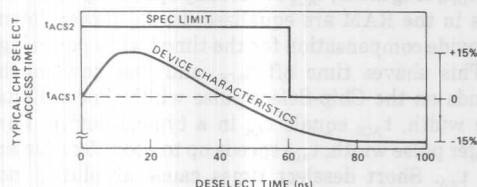
As t_{DPW} lengthens, t_{ACS} eventually speeds up. Certain nodes in the RAM are equalized during power-down to provide compensation for the time lost in powering up. This shaves time off t_{ACS} , and just how much depends on the Chip-Select pulse width. For a 40-ns pulse width, t_{ACS} equals t_{AA} in a typical device. For a longer pulse width, t_{ACS} speeds up to about 5 ns faster than t_{AA} . Short deselect times cause absolutely no problem for the device, but the slight increase in Chip-Select Access time must be allowed for. The device specifications account for this characteristic by speci-



3. When the auto power-down goes into effect, the 2147's current requirement drops as sections of the internal array are turned off.



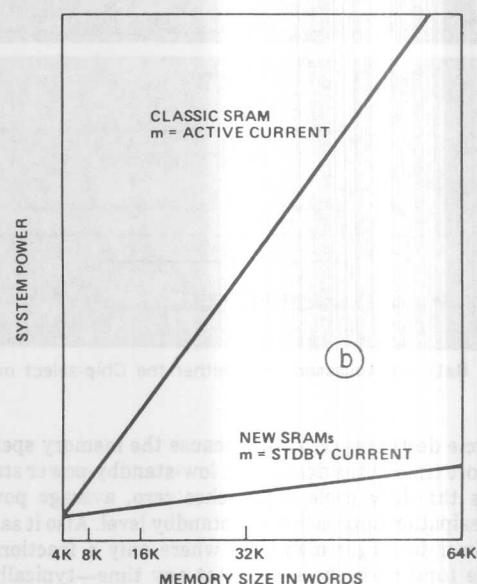
4. As their duty cycle decreases, the RAMs' power requirements drop; they spend more time in the deselected state



5. One major timing consideration—the relationship of Chip-Select access time to Deselect pulse width—must be dealt with. For deselect pulse widths of less than a cycle time, the access time typically increases by 5 ns.

Manufacturers of 4-k static RAMs

INTEL	Santa Clara, CA
AMD	Sunnyvale, CA
AMI	Cupertino, CA
EMM/SEMI	Phoenix, AZ
FUJITSU	Santa Clara, CA
INTERSIL	Cupertino, CA
MOSTEK	Carrollton, TX
MOTOROLA	Austin, TX
NATIONAL	Santa Clara, CA
NEC	Wellesley, MA
SIGNETICS	Sunnyvale, CA
SYNERTEK	Santa Clara, CA
T.I.	Houston, TX
TOSHIBA	Chicago, IL
ZILOG	Cupertino, CA

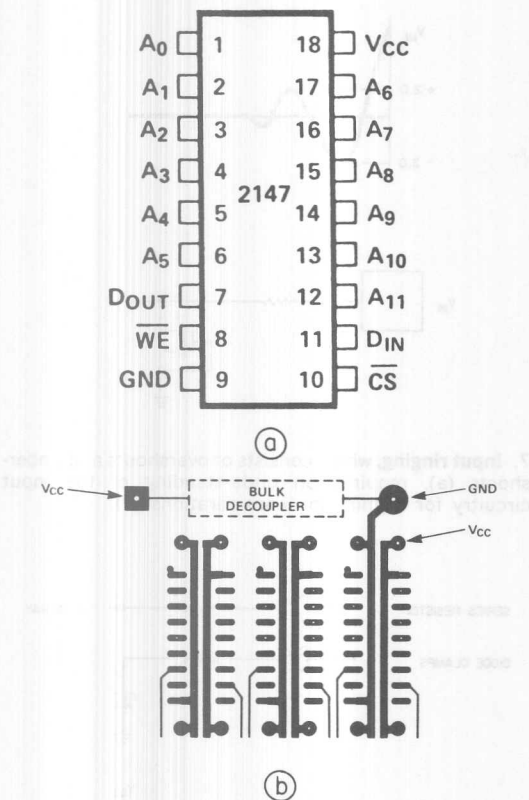


(a). Then only one bank of 2147s remains on at one time, allowing large systems substantial power savings (b).

fying two Chip-Select Access times, t_{ACS1} and t_{ACS2} .

The 2147's pinout, which follows the industry standard, was chosen for optimal performance and layout. All manufacturers but one have selected this pinout (or a clocked variation) for their medium and high performance 4-k \times 1 static RAMs (see Table).

The device's simple and efficient layout places the V_{CC} and ground at the corners, which simplifies routing and decoupling of the supplies (Fig. 6a). The address pins are placed to allow the address lines to be routed together, as are the data and control lines.



6. The 2147's pinout keeps V_{CC} and ground on the corners (a). The PC layout keeps the supply voltage fairly constant across a two-layer board (b).

The pinout minimizes cross-coupling effects by using the data pins to separate the control and address lines. This separation reduces Miller-capacitance effects between the address and control lines, whose signals frequently make near-simultaneous transitions. The interconnection cross-coupling from the data lines to either the address or control lines is minimal because of the usual perpendicular routing of the data traces to other traces.

The gridding used in the layout of Fig. 6b runs supplies both horizontally and vertically at every device location. This highly recommended gridding—in conjunction with the decoupling previously mentioned—keeps the supply voltage acceptably constant across a two-layer PC board.

Since RAMs like the 2147 operate in the high-speed world that was previously reserved for bipolar devices, line terminations are sometimes required to eliminate excessive overshoots, and the problem of ringing inputs must be faced. MOS inputs, which differ from bipolar inputs, do not provide input diode clamps. The 2147 incorporates an input protection circuit (Fig. 7) in which an n+ diffused resistor is used to limit

current transients from static discharge. The protection device provides an enhanced junction breakdown voltage. The diffused resistor forms a diode to the substrate.

Positive overshoots during a V_{IH} transition are no problem for the 2147. These overshoots seldom exceed the maximum input specification, and the levels that result from the ringing generally remain above the 2147's input threshold voltage of, typically, 1.5 V.

Negative overshoots during a V_{IL} transition also present no problem. To have an effect, the overshoot must be sufficiently negative to forward bias the device's input diode. This requires the overshoot to be 0.6 V more negative than the substrate (V_{BB} = -3 V typically), and it must last more than 20 ns—the input diodes approximate turn-on time. In the majority of designs, these conditions are not met. The diode is therefore not forward biased, and the overshoot has no effect.

In cases where the diode is forward biased, the small charge in the overshoot is injected into the floating substrate, slightly increasing the back-bias. As a result, device threshold voltages are raised and junction capacitances are decreased. Since these two changes have opposite effects on access time, the net change is limited to at least one or two nanoseconds (usually faster). The injected charge does not affect device reliability.

Negative overshoot can cause a problem if the subsequent ringing exceeds the input threshold voltage. If this occurs, the input buffer's reaction slows down as it tries to follow the changing input. This can lengthen a Read Access time by the time it takes the input lines to quiet down. For a Write cycle, addresses should be stabilized prior to an active Write pulse to eliminate the possibility of multiple selection and problems with stored data.

Terminations cut ringing

To avoid excessive ringing, use any of the terminations shown in Fig. 8. The series resistor is easiest, but it costs a few nanoseconds of performance. A typical resistor value is 10 to 33 Ω for high-speed Schottky gates. The parallel resistor network does not cost performance when matched to the line impedance, but does draw considerable power and imposes loading factors on the bus drivers that feed the RAMs. A Thevenin equivalent resistance of 100 to 200 Ω is typical for two-layer boards; it's less for multi-layer boards.

The R-C network saves power over the parallel resistor network, depending on frequency. Choose the resistor to match the impedance of the line (100 to 200 Ω). The capacitor should have about one-tenth the impedance of the resistor (i.e. $C = 10/2\pi fR$, about 30 to 150 pF). A simple Schottky diode clamp may also be considered.

During a negative overshoot, the protection device is too slow to have much of a clamping effect.

However, when the input is held at a dc negative level exceeding about -1 V, the protection device turns on and supplies current from ground to the input. Depending on the input voltage, the current can go as high as several milliamps.

System power-on does not immediately activate the 2147 back-bias generator. It begins functioning only when the V_{CC} supply has reached approximately 2.5 V. During this interval, device current can exceed standby specifications because internal threshold voltages are lower without back-bias, and device currents are consequently higher. The amount of current depends on whether the device is deselected (\overline{CS} high) or selected (\overline{CS} low).

If the device is selected, the power-on current rises quickly toward full active power (140 to 180 mA), as shown in Fig. 9a. Although no problem for the device, which has been designed to handle this current level, this current can cause a problem for the power supply. The supply was designed to handle current at or near device standby current levels, such as in a large memory application.

Deselect RAMs to keep power down

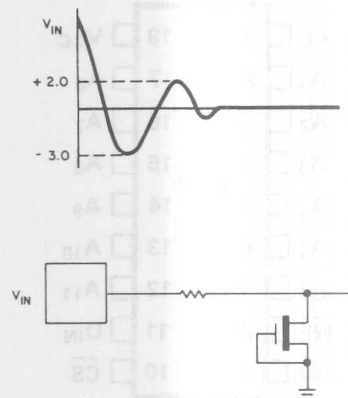
An obvious way to eliminate this problem is to keep the devices deselected during power-on. Simply use 1-k pull-up resistors to V_{CC} on the \overline{CS} inputs, which raises \overline{CS} as the power comes on. This holds the power-on current to about twice the standby current level (I_{SB}), which is considerably less than full active current (see Fig. 9b). However, this is still not quite as low as I_{SB} , and the power supply must be designed accordingly. A maximum power-on current spec (I_{PO}) is included in the 2147 data sheet for this purpose. Device current values range from 30 to 70 mA, depending on the version selected.

The time constant of the internal back-bias generator is about 10 to 100 μs —several times faster than most power supplies, whose constant is typically several milliseconds. Therefore, the dc curve of Fig. 9b represents what can be expected of a 2147 during ac power-up. The time spent within a specific voltage range will be determined by the time constant of the power supply—not the RAM.

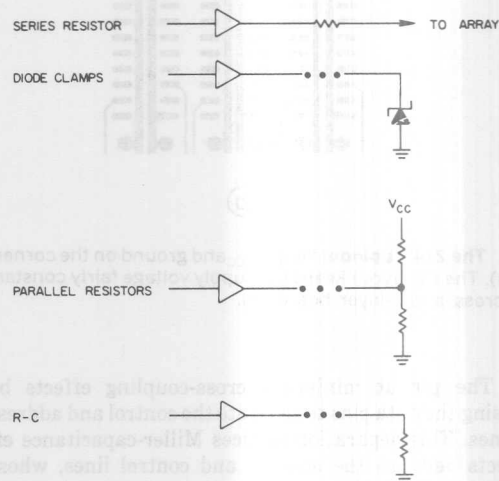
Putting the RAMs into a system using the suggested guidelines is relatively simple. Fig. 10a shows the basic block diagram for a 16-kword memory card using 64 of the 2147 (55 ns) RAMs. The card is designed to interface to a system that has an 18-bit address bus, a 16-bit data bus and a multi-line control bus.

The data bus can be organized as either a common I/O bus of 16 lines or a separate input and output bus totalling 32 lines. Within the card, data input and output lines are separate. Data written into the memory are latched and buffered by latches. The same goes for data read out of the memory.

Since the memory is static, the control bus consists of only two lines—a \overline{REQ} line that initiates each memory cycle, and a \overline{WR} line that defines whether



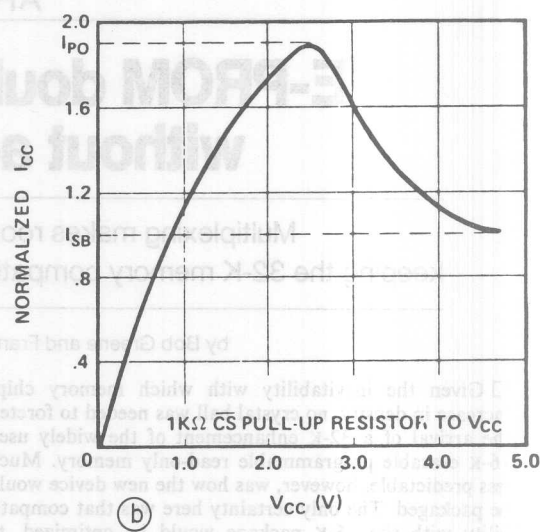
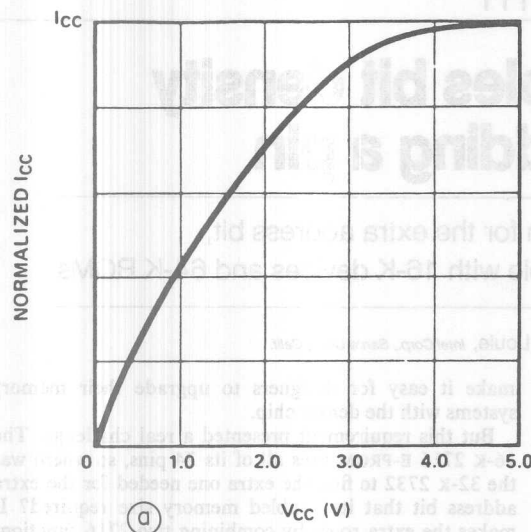
7. **Input ringing**, which consists of overshoots and undershoots (a), requires an understanding of the input circuitry for termination considerations (b).



8. To prevent excessive ringing, simple termination schemes such as these can be used. However, depending upon the specific system limitations, no single technique is always the ideal solution.

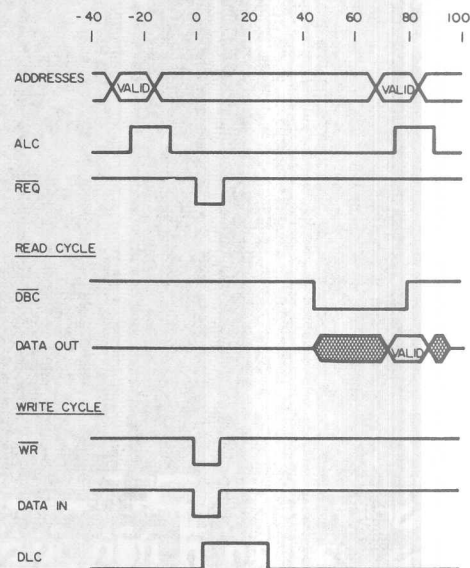
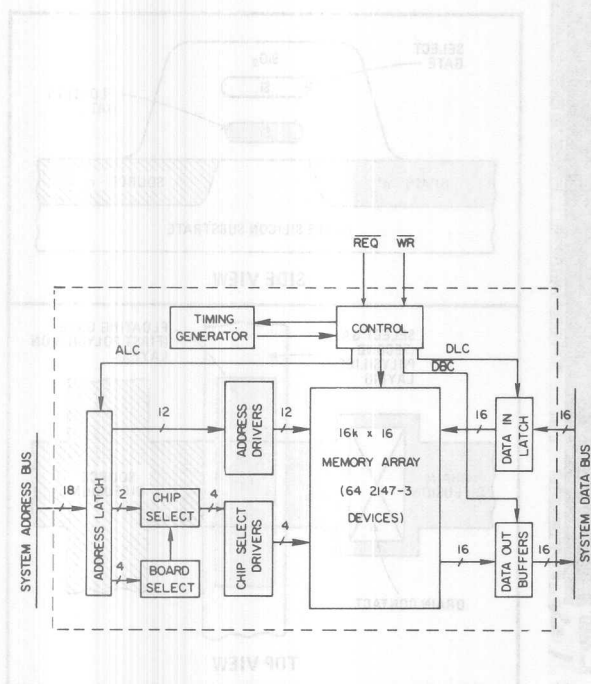
a read or write cycle will take place. Simplified timing waveforms for the system are shown in Fig. 10b. Addresses are established 30 ns prior to a cycle and latched by the time \overline{REQ} initiates the cycle at $t = 0$ ns.

The status of \overline{WR} at the beginning of the cycle determines whether a Read (\overline{WR} high) or Write (\overline{WR} low) is executed. If a Read is executed, data are available at the card edge at $t = 75$ ns. If a Write is executed, the input data are latched at the beginning of the cycle and the write is completed during the cycle. The maximum time required to complete either a Read or Write operation is 100 ns.■



9. When a RAM system is first powered up, the supply current can approach a fully active power level (a). During

power on, though, all RAMs can be kept deselected to minimize the start-up current from the power supply (b).



10. A typical, 16-kword (16-bit words) static RAM system is based on the 2147-3 (a). This memory system provides

an over-all cycle or access time of 100 ns, as shown by the timing waveforms (b).

without ^{AR-111} adding a pin

Multiplexing makes room for the extra address bit, keeping the 32-K memory compatible with 16-K devices and 64-K ROMs

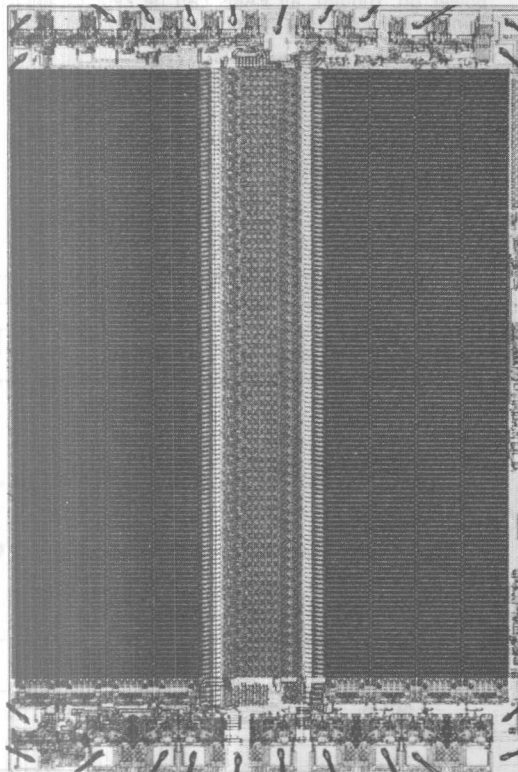
by Bob Greene and Frank Louie, Intel Corp., Santa Clara, Calif.

□ Given the inevitability with which memory chips increase in density, no crystal ball was needed to foretell the arrival of a 32-K enhancement of the widely used 16-K erasable programmable read-only memory. Much less predictable, however, was how the new device would be packaged. The only certainty here was that compatibility with the 16-K package would be optimized, to

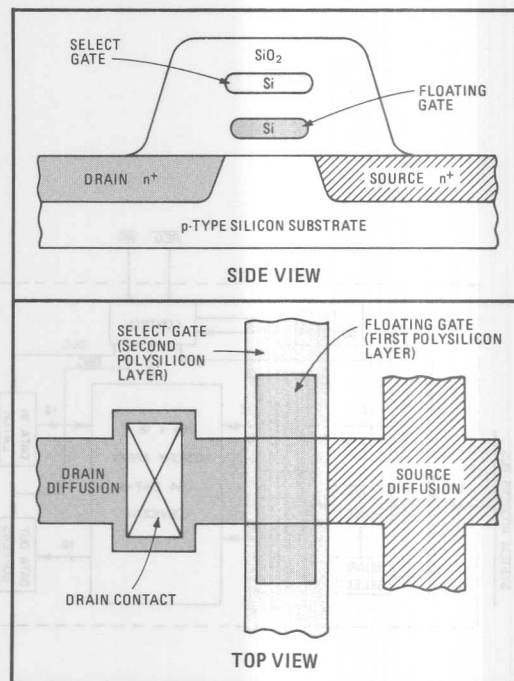
make it easy for designers to upgrade their memory systems with the denser chip.

But this requirement presented a real challenge. The 16-K 2716 E-PROM uses all of its 24 pins, so where was the 32-K 2732 to find the extra one needed for the extra address bit that its doubled memory size required? It makes the extra room by combining two 2716 functions on a single pin.

The final 2732 package truly meets the compatibility requirement. It can be used to build a memory board that is flexible enough to allow any mix of the read-only memory chips while affording a clear-cut modularity

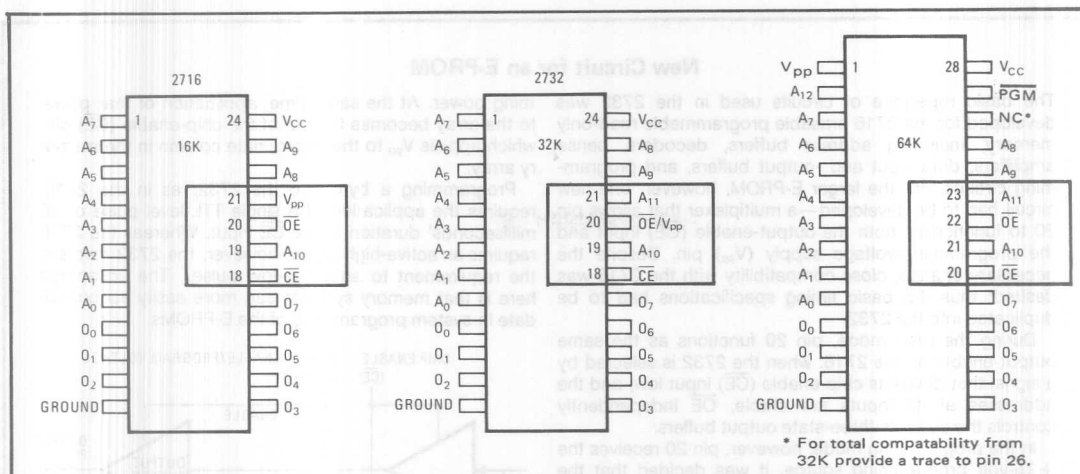


1. Four kilobytes of E-PROM. The 2732 erasable programmable read-only memory packs 32,768 bits into a chip measuring less than 40,000 mil², an area that is only 30% larger than the earlier 16-K 2716 memory chip. Placing all the bonding pads at two ends of the die, rather than around the whole perimeter, also helps boost density.



2. Tighter cell. The 2732 uses the same basic double-polysilicon stacked-gate MOS cell as its 16-K forerunner, although the 32-K chip is only 30% larger. Greater density of the new E-PROM derives from improvements in layout and photolithographic techniques.

Electronics/August 16, 1979



3. New pinout. To accommodate the extra address bit (A₁₁), the 2732 multiplexes the output-enable ($\overline{\text{OE}}$) with the programming-voltage input (V_{pp}) on pin 20. Compatibility is assured with next-generation 64-K E-PROM, which will be housed in a 28-pin package.

scheme for varying the page sizes and boundaries within a memory system.

The 2732 puts 32,768 bits of ultraviolet-light-erasable programmable memory on a chip less than 40,000 square mils in area (Fig. 1). It therefore packs twice the bits of the 2716 onto a chip only 30% larger. One reason is that all the bonding pads are on two opposing sides of the die, rather than around its entire periphery, as on the 2716. The advantage is an increase in the relative density of many of the circuits peripheral to the chip's actual memory array that would not be possible were those buffers and control circuits strung out around the chip perimeter.

In addition, improvements in circuit layout and photo-lithography have contracted the size of the die, even though the 32-K part is fabricated with the same two-level polysilicon stacked-gate MOS process as the 2716. In essence, the 2732 uses the same basic cell as the 2716 (Fig. 2).

The power dissipation of the 2732, which operates with a single +5-volt supply, is a maximum of 750 milliwatts—50% more power than the 2716. Like the 2716, however, the chip goes automatically into a standby mode when not selected, reducing its dissipation to the much lower value of 150 mW. That arrangement saves 80% of the power while not degrading system speed in the least—the access time of the device is 450 nanoseconds in the worst case.

Enter multiplexing

For compatibility with the 16-K E-PROM, the 2732 maintains the same two control lines: an output-enable input ($\overline{\text{OE}}$), which independently controls the chip's three-state output buffers, and a chip-enable input ($\overline{\text{CE}}$), which selects the device and provides the automatic power-down feature.

So that the 2732 would fit into the same 24-pin package as the 16-K E-PROM, the new twelfth address bit (A₁₁) is given the programming-voltage supply (V_{pp}) pin,

while V_{pp} and $\overline{\text{OE}}$ are now multiplexed to share a single pin, as shown in Fig. 3. The multiplexing relies on a voltage-dependence scheme that is transparent to the user operating the chip in its normal read mode (see "New circuit for an E-PROM," p. 128).

System applications

With the new pin arrangement, an extremely flexible memory system can be planned around 28-pin package sites that allows the page size—the number of bytes per site—to change easily from 1 kilobyte to 8 kilobytes. (Although the largest increment, a 65,536-bit device, is available now only as masked ROM, the 64-K E-PROM will soon join the family.)

The 2732, with a 4-kilobyte capacity, is well suited to many microprocessor program-storage applications. Perhaps more important, the 2732 design and pinout allow a new degree of modularity with respect to system page size—the universal-board concept is closer than ever, for page sizes of 1, 2, 4, and 8 kilobytes can be designed into a system, and when the system is configured at the time of card assembly, the correct ROM or E-PROM can be inserted in the sockets provided. Moreover, the output- and chip-enable lines completely eliminate bus contention and keep the system operating at a minimum power dissipation.

Architecture for a flexible board

While it is generally true that the average ratio of read-only to read/write memory in a microprocessor-based system ranges from 3:1 to 5:1, systems often are of necessity committed to a hardware design well before the exact amount of RAM and ROM required has been established. But with a little advance planning, it is possible to execute a scheme that permits page size to vary by allowing the ratio of RAM to ROM to be decided after the hardware is built.

The key to such an architecture is a fuse-link-programmable ROM for address decoding. All that is needed

New Circuit for an E-PROM

The basic repertoire of circuits used in the 2732 was developed for the 2716 erasable programmable read-only memory, including address buffers, decoders, sense amplifiers, data-input and -output buffers, and programming circuits. For the larger E-PROM, however, one new circuit had to be developed—a multiplexer that allows pin 20 to function as both the output-enable (\overline{OE}) input and the programming-voltage supply (V_{pp}) pin. Despite the necessary change, close compatibility with the 2716 was desired; thus the basic timing specifications had to be duplicated into the 2732.

During the read mode, pin 20 functions as the same output-enable on the 2716: when the 2732 is selected by a signal that drives its chip-enable (\overline{CE}) input low, and the addresses at its inputs are stable, \overline{OE} independently controls the device's three-state output buffers.

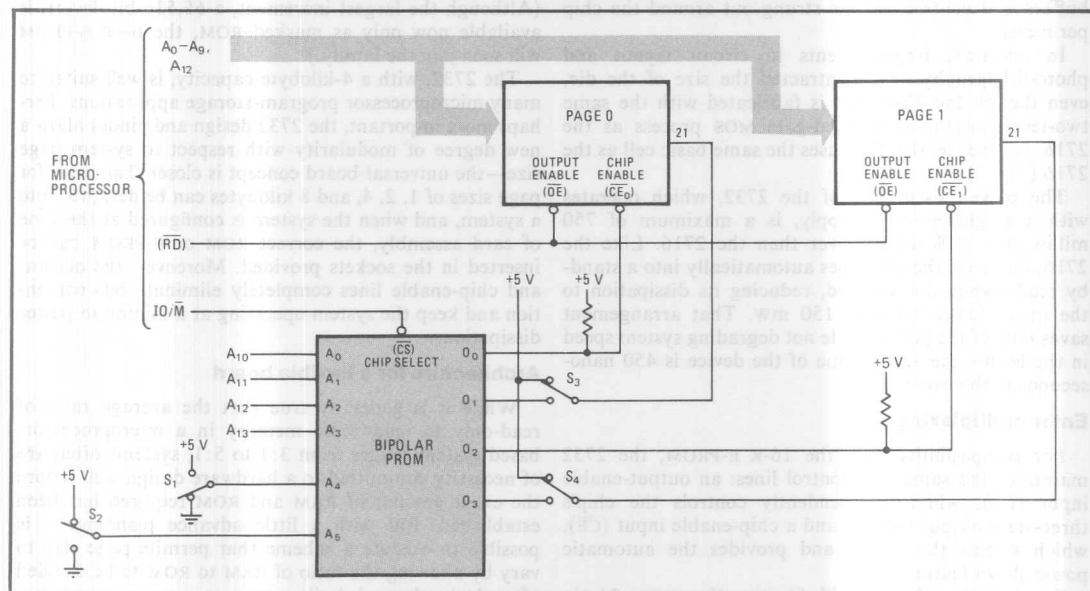
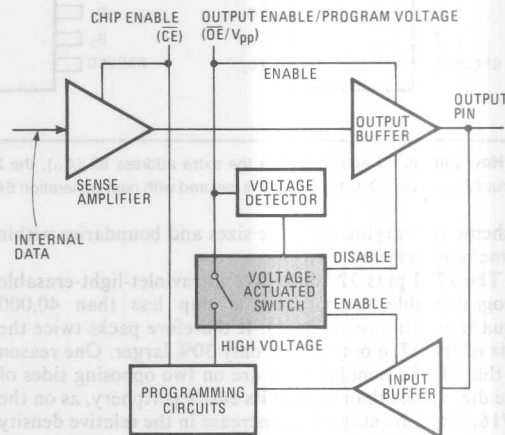
In the programming mode, however, pin 20 receives the +25-volt programming source. It was decided that the mere presence of that high voltage would switch the 2732 into the programming mode, which requires reversing the data outputs so that they become inputs, and switching the function of the chip-enable input to that of a programming-enable input.

The problem was in keeping the switch to the programming mode transparent to the chip's normal read operation. The solution for pin 20 was the voltage-activated switch whose circuit is diagrammed in the figure.

When a high voltage (V_{pp}) is detected at the pin, the output buffer is disabled and the programming-input buffer enabled, thereby turning the data outputs around so that they become inputs. The switch also allows that pin 20 become the source of the relatively high program-

ming power. At the same time, application of that power to the array becomes the job of the chip-enable (\overline{CE}) pin, which applies V_{pp} to the appropriate column in the memory array.

Programming a byte into the 2732, as in the 2716, requires the application of a single TTL-level pulse of 50 milliseconds' duration to the \overline{CE} input. Whereas the 2716 requires an active-high signal, however, the 2732 reverses the requirement to an active-low pulse. The advantage here is that memory systems can more easily accommodate in-system programming of the E-PROMs.



4. Variable-density pages. The key to a flexible memory board with variable page size—the number of bytes per socket—is a fuse-link PROM for decoding. Adding dual in-line switches lets page sizes and boundaries be set in the field after software is finalized.

	Page boundary for 1-kilobyte page	Page boundary for 2-kilobyte page	Page boundary for 4-kilobyte page	Page boundary for 8-kilobyte page		
	A ₁₀	A ₁₁	A ₁₂	A ₁₃	A ₁₄	A ₁₅
64-K ...						
16-K	0	0	0	0	1	0
15-K	1	1	1	1	0	0
14-K	0	1	1	1	0	0
13-K	1	0	1	1	0	0
12-K	0	0	1	1	0	0
11-K	1	1	0	1	0	0
10-K	0	1	0	1	0	0
9-K	1	0	0	1	0	0
8-K	0	0	0	1	0	0
7-K	1	1	1	0	0	0
6-K	0	1	1	0	0	0
5-K	1	0	1	0	0	0
4-K	0	0	1	0	0	0
3-K	1	1	0	0	0	0
2-K	0	1	0	0	0	0
1-K	1	0	0	0	0	0

for a basic system is a four-output PROM, such as a 1Kx4 bipolar device. (A 1,024-by-4-bit device is used in this case; it is large enough to accommodate a universal coding scheme with many more combinations of page sizes and boundaries.)

As shown in the simplified two-page example of Fig. 4, the 10 least significant address bits (A₀-A₉, which address a 1-kilobyte space) are passed through the system and connect directly to pin addresses A₀-A₉ at all the memory sites. Bits A₁₀-A₁₃ go to the PROM's least significant address inputs A₀-A₃. (In this example only a 16-kilobyte space is addressed; no use is made of the most significant address bits A₁₄ and A₁₅ that reach the 64-K byte address space found in most microprocessors.)

For an 8085 or 8086 microprocessor, the IO/Mpin, which determines whether the processor reads from the input/output lines or from memory, connects to the chip-select input (\overline{CS}) of the PROM. The microprocessor's read (\overline{RD}) signal (or the MRDC signal on an 8288 bus-controller used in conjunction with an 8086) drives the \overline{OE} signals of all the memories in the system.

Table 1 shows the address-memory map for determining the page sizes and boundaries, which are fixed by the PROM. Note that the system address bit that controls the page boundary changes as the page size is changed. For the 4-kilobyte 2732, for example, system address bit A₁₂ determines the page boundary, while if a 2-kilobyte 2716 were used, system address bit A₁₁ would determine the

System address	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀	CE ₀	Pin 21	CE ₁	Pin 21
Decoder programmable read-only memory										
1 kilobyte/page	0	0	0	0	0	0	0	X	1	X
	0	0	0	0	0	1	1	X	0	X
2 kilobytes/page	1	0	0	0	0	0	0	X	1	X
	1	0	0	0	0	1	0	X	1	X
	1	0	0	0	1	0	1	X	0	X
	1	0	0	0	1	1	1	X	0	X
4 kilobytes/page	0	1	0	0	0	0	0	0	1	0
	0	1	0	0	0	1	0	0	1	0
	0	1	0	0	1	0	0	1	1	1
	0	1	0	0	1	1	0	1	1	1
	0	1	0	1	0	0	1	0	0	0
	0	1	0	1	0	1	1	0	0	0
	0	1	0	1	1	0	1	1	0	1
	0	1	0	1	1	1	1	1	0	1
8 kilobytes/page	1	1	0	0	0	0	0	0	1	0
	1	1	0	0	0	1	0	0	1	0
	1	1	0	0	1	0	0	1	1	1
	1	1	0	0	1	1	0	1	1	1
	1	1	0	1	0	0	0	0	1	0
	1	1	0	1	0	1	0	0	1	0
	1	1	0	1	1	0	0	1	1	1
	1	1	0	1	1	1	0	1	1	1
	1	1	1	0	0	0	1	0	0	0
	1	1	1	0	0	1	1	0	0	0
	1	1	1	0	1	0	1	1	0	1
	1	1	1	0	1	1	1	1	0	1
	1	1	1	1	0	0	1	0	0	0
	1	1	1	1	0	1	1	0	0	0
	1	1	1	1	1	0	1	1	0	1
	1	1	1	1	1	1	1	1	0	1

Note: All unused decoder PROM address inputs should be tied to ground. X = V_{CC} via switch S₃ or S₄.

boundary. (It is important in using 2716s to remember that pin 21 must be tied to +5 v.) In a similar manner, if 8-kilobyte 64-K ROMs or E-PROMs were used, system address bit A₁₃ would determine the page boundary.

Coding the PROM

The PROM is coded so that its address bits A₄ and A₅ select the page size, which is determined in this case by switches S₁ and S₂. The switching scheme is as follows: for a 1-kilobyte page, A₄ and A₅ are both 0; for a 2-kilobyte page, they are 0 and 1, respectively; for a 4-kilobyte page, they are 1 and 0; and for an 8-kilobyte page, both A₄ and A₅ are 1.

The entire code for the PROM is shown in Table 2. By utilizing additional switches and inputs to the PROM, the various combinations of page size can be provided. If desired, a universal code may be developed so that one PROM may accommodate any changes in page size. Ultimately, the unused address bits of the PROM could be utilized to allow the various page sizes to be assigned anywhere on the memory map, and single-pole, double-throw switches (the dual in-line package type) could allow the page configuration to be changed in the field to suit the software. □

AR-112A UNIVERSAL BYTE WIDE PINOUT: 2764 IS THE KEY

FOREWARD

NOTE: The following paper was presented to a recent JEDEC meeting which dealt with compatibility of EPROM memory pinouts and functions.

To cover this subject in detail it was necessary to reference certain unannounced Intel products. The mention of these products should not be construed as an announcement or in any way imply product availability.

ABSTRACT

This paper describes a compatible pinout family that encompasses several classes of memory devices which fit in a standard 24 pin and 28 pin DIP site. This pinout format is equally suitable for ultraviolet erasable PROMs (EPROM), Mask ROMs, Electrically Erasable PROMs (E²), byte wide Static RAMs, and byte wide dynamic or pseudo-static RAMs.

BACKGROUND

For several years, system designers have had available a compatible family of EPROMs from Intel that allows one kilobyte, two kilobyte or four kilobyte devices to be used interchangeably in the same socket. By anticipating the 64K (8K X 8) EPROM and using 28 pin sites, this compatible family can be extended to utilize all 28 pin devices in a single format. There are JEDEC standards which govern the pinout of the 32K EPROMs, but the approval of a dual pinout standard seems to have confused prospective users seeking family and class compatibility from the various manufacturers. These various classes of devices, which include Electrically Erasable (E²), byte wide Static RAMs and pseudo-static RAMs, will be available from Intel and other manufacturers during 1980.

Before proceeding two key definitions of system required functions are in order:

\overline{CE} (active Low Chip Enable) is located on pin 18 of the 2716, 2732 and 2732A devices. Its operation in a system is to perform the power up function in the device. The \overline{CE} function is the primary control function; it is uniquely decoded from a particular pattern of system addresses. Utilizing the \overline{CE} function in this manner allows all other non-selected devices in a system to be in their low power mode.

\overline{OE} (active Low Output Enable) is located on pin 20 of the present devices. Its function in a system is to provide an independent control over the output buffers internal to the memory device, thereby allowing the READ signal from the microprocessor to be connected to all \overline{OE} s present in the entire memory array. In this way, the data bus is only active when required by the processor or, more exactly, when the processor requires or expects data from the memory device that was selected via the \overline{CE} function.

The two control lines are ANDed inside the device; this means that only the coincident application of \overline{CE} and \overline{OE} will activate the output of the memory device—the application of \overline{OE} alone will not cause the outputs to change from the high impedance state.

It is Intel's belief that the use of an independent output enable is the only way of assuring that there is no bus contention in a system. The use of non integrated output buffers cannot achieve the same result; they can only confine bus contention to a memory card or memory section of a large card. In addition, as processor speeds increase, greater demands are placed on memory performance—the use of external non integrated output buffers places still more requirements on the performance of the memory. In this context, the time between addresses out and data in is a fixed period of time for any given processor—all devices inserted in the path—demultiplexers, transceivers, decoders, etc., must be offset by higher performance memory speeds.

The pinout family described in this paper incorporates not only maximum flexibility for the system designer with respect to the various densities of devices that are available but also allows the freedom of flexible boundaries within the memory map for different classes of memories, after the printed circuit board and system are manufactured.

Furthermore, this pinout family maintains the system control features required for functionality as densities progress from the 32K level to the 64K level, while, if the 64K device is "squeezed" into a 24 pin package, one control pin must be given up to provide for the additional required address.

FUNCTIONALITY REQUIRED IN A UNIVERSAL PINOUT

The overall objective of Intel's compatible pinout format is to provide all of the system functions required by any memory class that may be inserted into this universal site. For this site to be truly universal, it must contain provisions for address lines that represent memory densities that have not yet been developed by semiconductor manufacturers.

The attached diagram shows the pinout of the 2764—the 8K X 8 EPROM. This is the pinout that is the key to the universal pinout. The system control pins (\overline{CE} and \overline{OE}) have already been discussed; the address and I/O pins remain standard both with respect to TTL compatibility and physical location. There are only 5 pins that need to be discussed and defined.

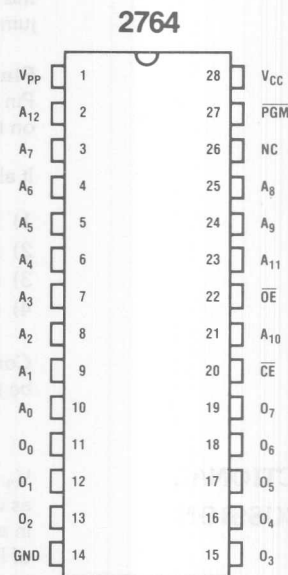


Figure 1.
The Intel 2764 (8K X 8 EPROM)
—Key to a Universal Pinout

Pin 1

Pin 1 serves as the system high voltage (V_{PP} is connected to V_{CC} for EPROM class devices). As we develop the future functionality of this pin, we will require it to serve as V_{PP} for the Write and Erase functions of the Electrically Erasable (E^2) devices, and it would also be very desirable for the 8K X 8 pseudo-static RAM to have the external refresh pin be located here—and implemented as RFSH, not \overline{RFSH} . (Present proposals call for RFSH.) In that way, when pin 1 is tied to V_{CC} , the RAM would be self refreshing. This makes sense from a system point of view—for use with

associated with each Γ in 1, any universal site could accommodate an E^2 device, an EPROM, or a pseudo-static RAM.

Pin 2

Pin 2 is the connection for system address A_{12} for 8K X 8 and larger devices. In addition it is a control connection for the 28E2 "smart" 2K X 8 E^2 devices. As with Pin 1, a single jumper or a jumper at each site could be used to allow the use of the 28E2 in an EPROM/ E^2 /RAM system. The actual jumper configuration will be system dependent that is—the use of "per site" jumpers may be more efficient in some systems than a single, card edge jumper. In 28E2 board usages, per site jumpers should be provided.

Pin 28

Pin 28 is the V_{CC} supply for all devices in the universal pinout. And, because all the members of the family have a \overline{CE} function, the current required by this pin will be reduced to approximately 25% of the maximum when the device is deselected.

Pin 27

Pin 27 is the writing function for all members of the universal pinout family. In the case of the EPROMs it is called PGM, while for the E^2 and pseudo-static RAMs it is \overline{WE} . When technology advances to the point of allowing a 256K (32K X 8) EPROM, this pin will become A_{14} . This pin is effectively accommodated with a card edge jumper.

Pin 26

Pin 26 (which corresponds physically to pin 24 of a 24 pin device) is a No Connect on the 2764.

It also must serve as:

- 1) V_{CC} for the various devices in 24 pin packages (2716, 2732A, 21R1, etc.)
- 2) A_{13} for the 27E4 (16K X 8 EPROM)
- 3) CNTRL for the 28E2 (2K X 8 E^2)
- 4) CNTRL for the 21R2 (8K X 8 RAM)

Consequently, this pin may be thought of as the "class configuration pin" as it must be jumpered either at the site or at the card edge to allow total flexibility.

FUNCTIONAL DISCUSSION

V_{PP} is the signal/power supply that is required for programming of EPROMs as well as writing into E^2 devices. Since EPROMs are not normally programmed or written to in a system environment, the V_{PP} supply is set equal to V_{CC} for read only applications of EPROMs. E^2 devices require an in-system V_{PP} supply in excess of 20 volts in order to write to the device in a manner similar to EPROMs. In keeping with the tradition of locating power supplies at the corners of device sites, V_{PP} is placed on pin 1. (Although not part of the functional discussion, ground is located on pin 14 and V_{CC} is located on pin 28.) To further discuss other functions that will be present on pin 1, the refresh signal for the 8K X 8 pseudo-static RAM is placed on pin 1. As mentioned above, pin 1 will normally be tied to V_{CC} for use in EPROM systems. The active high refresh control being proposed as a standard would allow an 8K X 8 pseudo-static RAM to be inserted into the same socket and the refresh function would be taken care of automatically, that is, the pseudo-static RAM is self-refreshing with the signal RFSH tied high. (Implementation of RFSH requires an additional logic element to accomplish automatic refresh.)

COMPATIBILITY WITH PRESENT DEVICES

The other functional control pin that needs to be discussed is write-enable (\overline{WE}), which is found on pin 27. This provides the write function for the pseudo-static RAM, the Static RAM, the E^2 PROM and the future class of non-volatile memory. Remembering that EPROMs are rarely programmed in a system environment; it is pin 27 that provides the \overline{PGM} function for the EPROM. In a multi-memory system it is the intention that all pin 27s would be tied together and connected to the source of write-enable from the microprocessor. In a normal system WRITE pin 27 will be taken LOW; with V_{PP} at 5 volts and \overline{CE} HIGH, no action will occur in an EPROM memory. To perform a write operation into an E^2 memory, V_{PP} must be greater than 20 volts, \overline{CE} must be LOW and \overline{WE} must be LOW. In the case where E^2 and EPROM memories are being used on the same card, and it is desired to write into the E^2 , the high voltage (V_{PP}) can be supplied to all devices and only that device which received \overline{CE} in proper timing with \overline{WE} will be written into.

Pin 26 is a no connection on the 8K X 8 EPROM. However, it is anticipated that most system designers will connect pin 26 to V_{CC} which allows the use of 24 pin devices in the lower 24 pins of the 28 pin site. For maximum flexibility pin 26 should be jumpered on the card edge to V_{CC} .

As the figure indicates, the 2732A pinout is a subset of the 28 pin universal pinout; the bottom 24 pins of the 2764 are identical to the 2732A. Likewise the 2716 is able to be inserted directly into the bottom 24 pins with the inclusion of a jumper to accommodate the V_{PP}/A_{11} changeover.

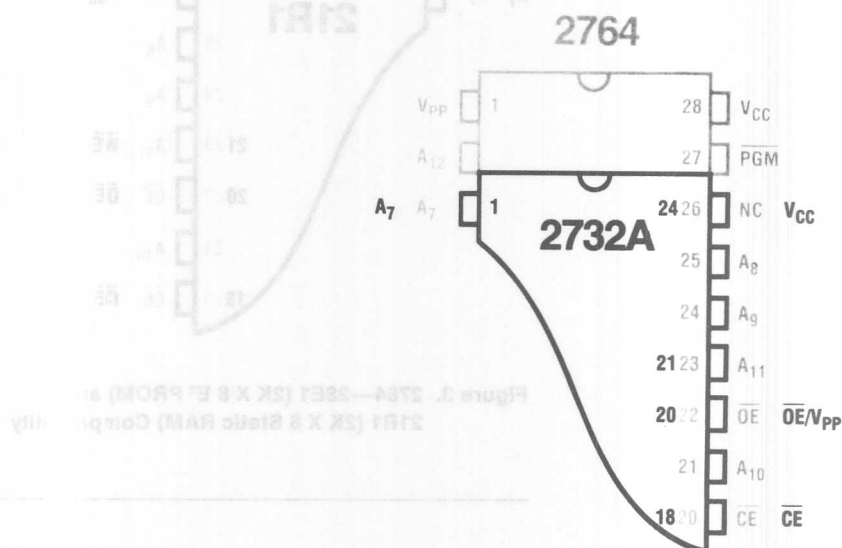


Figure 2. 2764—2732A (4K X 8 EPROM) Compatibility

In a similar manner, the 28E1 (2K X 8 E²) and the 21R1 (2K X 8 Static RAM) can be inserted in the lower 24 pins. System implementation of either of these devices require that pin 23 be appropriately jumpered—the E² device requires V_{pp} on pin 23 (pin 21 of the 24 pin device) while the RAM requires WE on that pin.

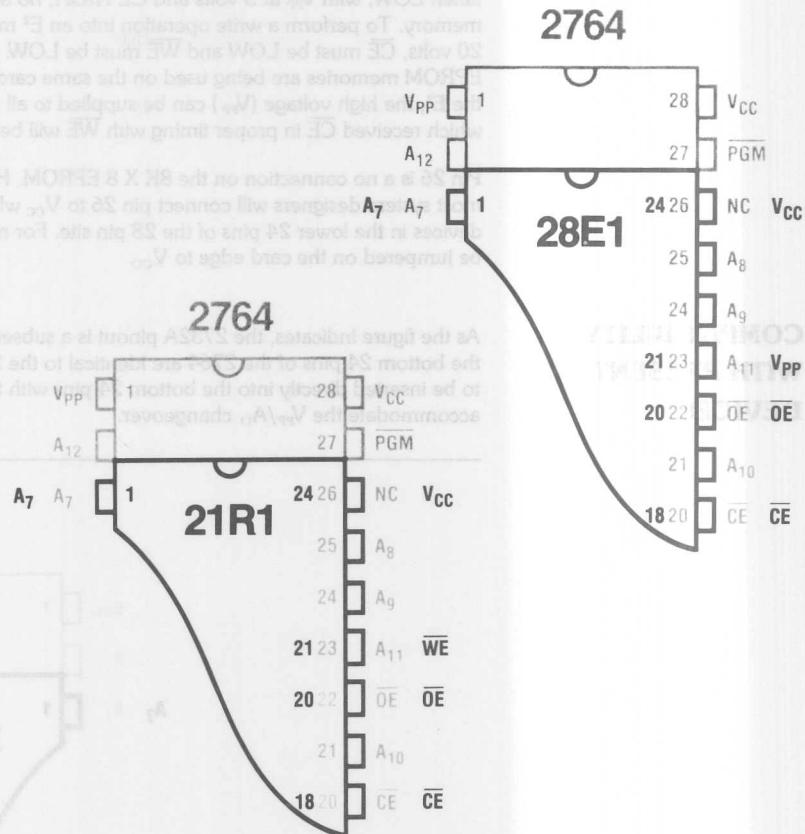


Figure 3. 2764—28E1 (2K X 8 E² PROM) and 21R1 (2K X 8 Static RAM) Compatibility

THE FUTURE

Some future devices that will fit the universal pinout are the 28E2 (2K X 8 "smart" E²) and the 21R2 (8K X 8 pseudo-static RAM). These devices require jumpers to accommodate their functionality—in the case of the 28E2, V_{pp} must be supplied to pin 1, pins 2 and 26 require control functions. This will require a jumper for pin 2 (it is also A₁₂ for 8K X 8 densities and above) while pin 26 could be hard wired if 24 pin devices are not anticipated to be used. A jumper on pin 26 will allow interchangeability with 24 pin devices, 16K X 8 EPROMs, smart 2K X 8 E² PROMs and 8K X 8 pseudo-static RAMs.

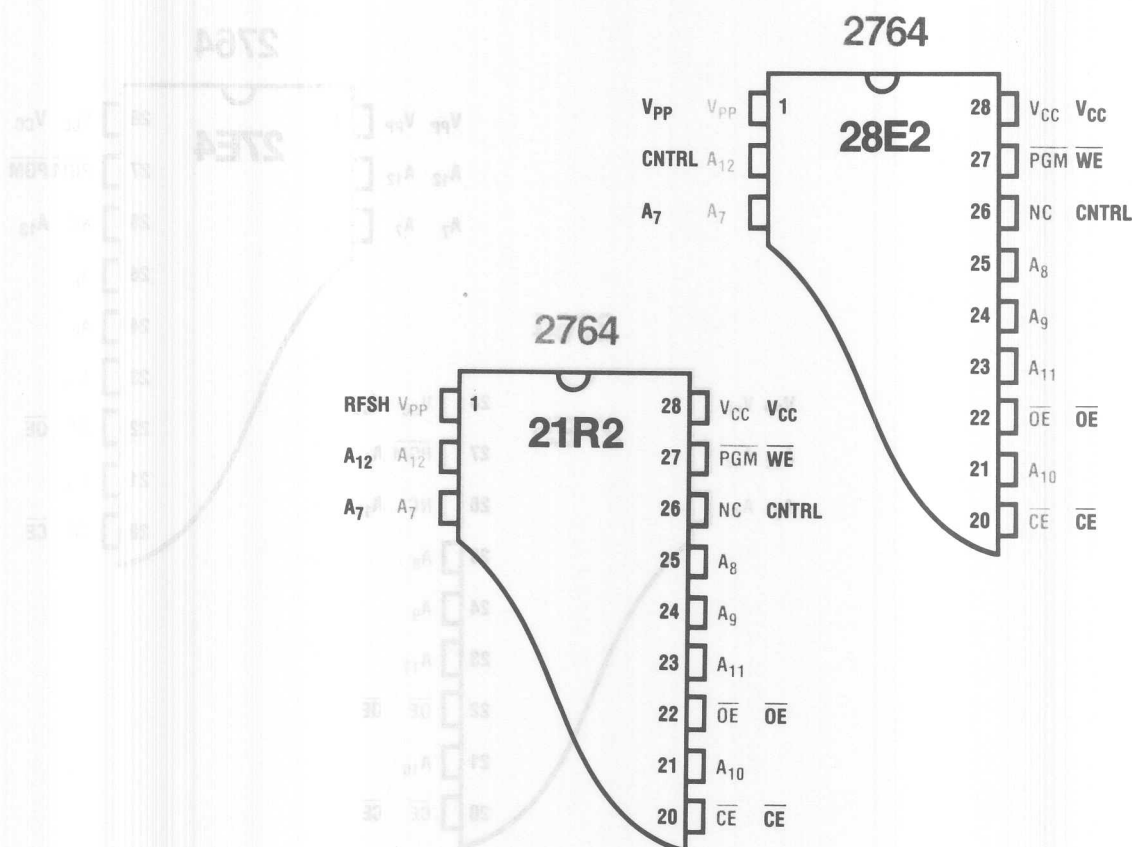


Figure 4. 2764—28E2 (2K X 8 Smart E² PROM) and 21R2 (8K X 8 Pseudo-Static RAM) Compatibility

the early eighties, it is necessary to anticipate the total compatibility that this proposed pinout scheme provides for the user. As can be seen in the attached figure, address A_{12} , which provides the 8 kilobyte capacity, is located on pin 2 of the 2764. The most logical place for address A_{13} , which is required for a 16 kilobyte device, is on pin 26. And finally, A_{14} will appear on pin 27, the last available pin.

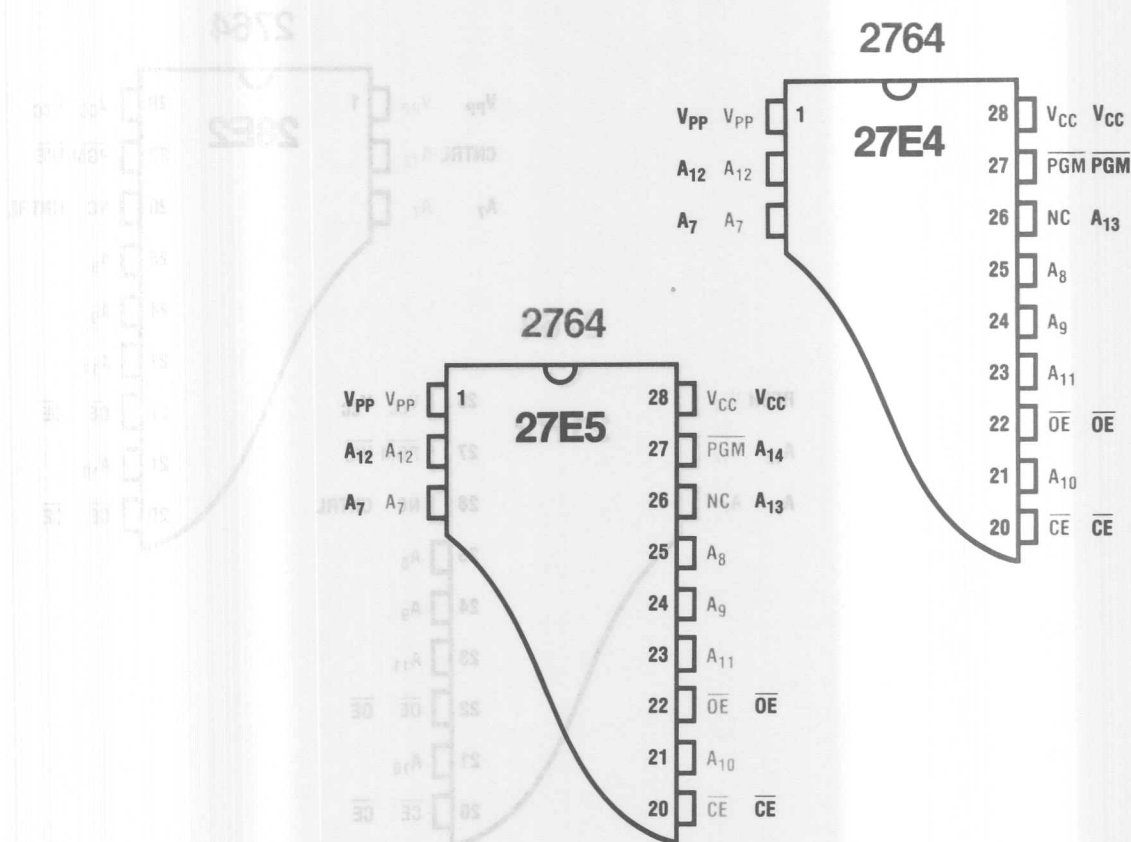


Figure 5. 2764—27E4 (16K X 8 EPROM) and 27E5 (32K X 8 EPROM) Compatibility

been demonstrated that this universal pinout provides compatibility with a total of 3 classes of devices and several densities of each of the devices in a given class. The pinout accomplishes the above compatibility with an absolute minimum number of jumpers, while maintaining functional compatibility with contemporary micro-processors. And it also is compatible with Intel's plan for a future class of non volatile memories.

For purposes of comparison all required jumpers are shown in the table below. The figures which follow summarize all the pinouts of the various classes and densities discussed in this paper.

PIN 1	PIN 2	PIN 26	PIN 27
+5V	A ₁₂	V _{CC}	WE (PGM)
V _{PP}	CNTRL	CNTRL	A ₁₄
RFSH (RFSH)		A ₁₃	

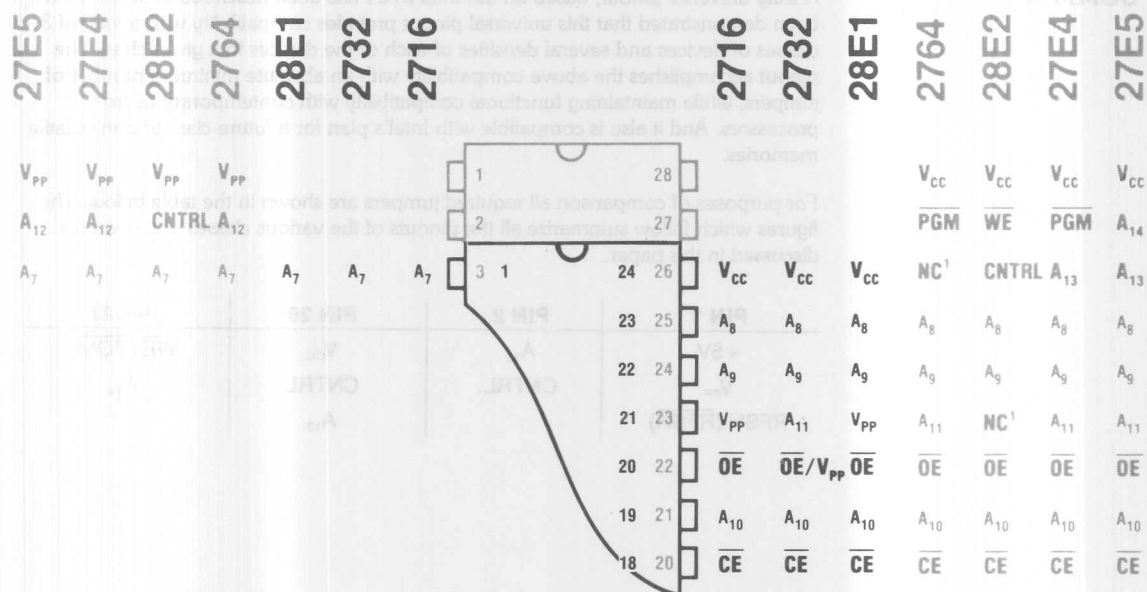
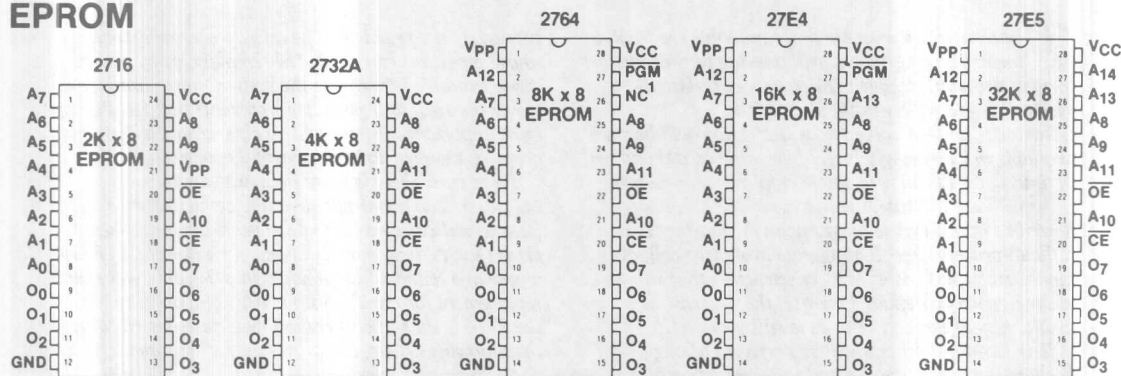


Figure 6. Pinout Summary—EPROM, E²

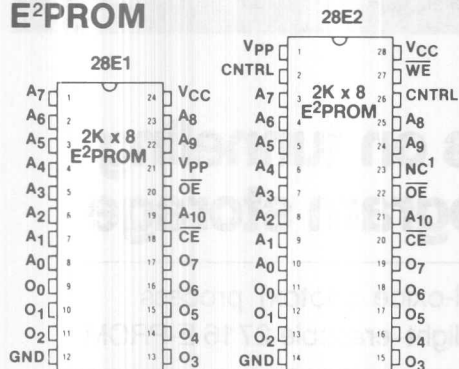
NOTE 1—

An address should be provided for upward compatibility.

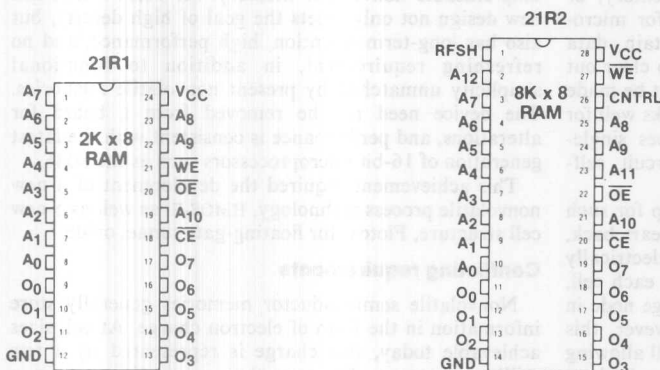
EPROM



E²PROM



RAM



NOTE 1

An address should be provided for upward compatibility.

Figure 7. Detailed Pinout Diagram by Class and Density

The electrically erasable programmable read-only memory, or EE-PROM, will one day be the standard form of program storage in microprocessor-based systems. It will follow in the steps of the ultraviolet-light-erasable PROM, for it, too, will become available in increasingly larger byte-wide arrays and will in time share silicon with single-chip microcomputers.

As with the E-PROM, the success of the EE-PROM described in this article hinges upon the mastery of a difficult process. The floating-gate avalanche cell, also pioneered by Intel, is a tricky construction that still eludes many a memory maker. Likewise, the widespread availability of large EE-PROMs is still years off.

The EE-PROM process will be perfected, though, because the rewards go beyond the elimination of the expensive quartz window on the E-PROM package. The electrically erasable memory will usher in systems

previously not practical. The microprocessor system whose programs can be altered remotely, as by phone, is one example. Another is the system that is immune to power outages, as it protects its contents in ROM. Perhaps most important, systems will be able to adjust their own program memory to environmental changes.

To be sure, there is more than one way to build an EE-PROM. The metal-nitride-oxide-semiconductor (MNOS) structure has served for years in modest-sized arrays for TV tuning applications, for example. In fact, a year ago Hitachi Ltd. announced a 2-K-by-8-bit MNOS replacement for the 2716 E-PROM. Compatibility with the 2716 is the impetus behind the device described in the following article, but it uses only silicon and its derivatives, plus metal. Also, in place of avalanche injection, which can injure a cell, electrons tunnel to and from a floating gate.

-John G. Posa

16-K EE-PROM relies on tunneling for byte-erasable program storage

Thin oxide is key to floating-gate tunnel-oxide (Flotox) process used in 2,048-by-8-bit replacement for UV-light-erasable 2716 E-PROM

by W. S. Johnson, G. L. Kuhn, A. L. Renninger, and G. Perlegos, Intel Corp., Santa Clara, Calif.

□ The erasable programmable read-only memory, or E-PROM, is the workhorse program memory for microprocessor-based systems. It is able to retain data for years, and it can be reprogrammed, but to clear out its contents for new data, ultraviolet light must be made to stream through its quartz window. This works well for many applications, but the technique foregoes single-byte—in favor of bulk—erasure and in-circuit self-modification schemes.

Electrical erasability is clearly the next step for such memories, but like ultraviolet erasure a few years back, it is hard to achieve. In fact, the design of an electrically erasable read-only memory is paradoxical. In each cell, charge must somehow be injected into a storage node in a matter of milliseconds. Once trapped, however, this charge may have to stay put for years while still allowing the cell to be read millions of times. Although these criteria are easily met individually, the combination makes for a design with conflicting requirements.

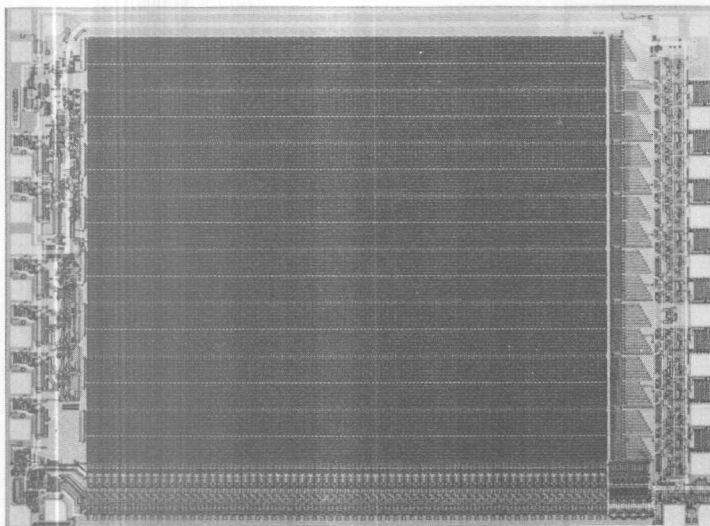
These demands are more than met in a new EE-PROM, which is a fully static, 2-K-by-8-bit, byte- or

chip-erasable nonvolatile memory. At 16,384 bits, this new design not only meets the goal of high density, but also has long-term retention, high performance, and no refreshing requirement, in addition to functional simplicity unmatched by present nonvolatile memories. The device need not be removed from a board for alterations, and performance is consistent with the latest generation of 16-bit microprocessors such as the 8086.

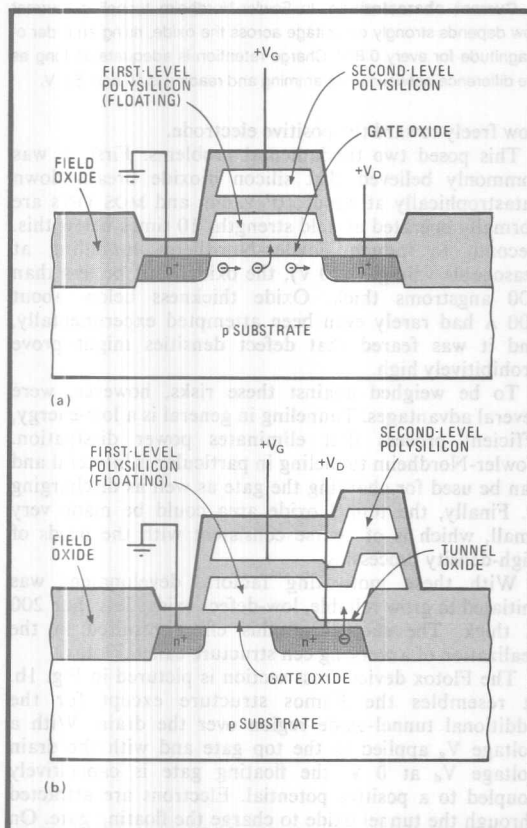
This achievement required the development of a new nonvolatile process technology, HMOS-E, as well as a new cell structure, Flotox, for floating-gate tunnel oxide.

Conflicting requirements

Nonvolatile semiconductor memories generally store information in the form of electron charge. At cell sizes achievable today, this charge is represented by a few million electrons. To store that many electrons in a 10-millisecond program cycle requires an average current on the order of 10^{-10} amperes. On the other hand, if it is essential that less than 10% of this charge leaks away in 10 years, then a leakage current on the order of



The next memory. The 16-K electrically erasable programmable read-only memory is eminently suitable for microprocessor program storage. Organized as 2,048 by 8 bits, the EE-PROM allows full-chip or individual-byte erasure using the same supply (V_{ee}) as for programming.



1. First Famos, now Flotox. The Famos cell (a) found in all E-PROMs stores charge on the floating gate by avalanche means. Flotox cell (b), the heart of the EE-PROM, relies on electron tunneling through thin oxide to charge and discharge the floating gate.

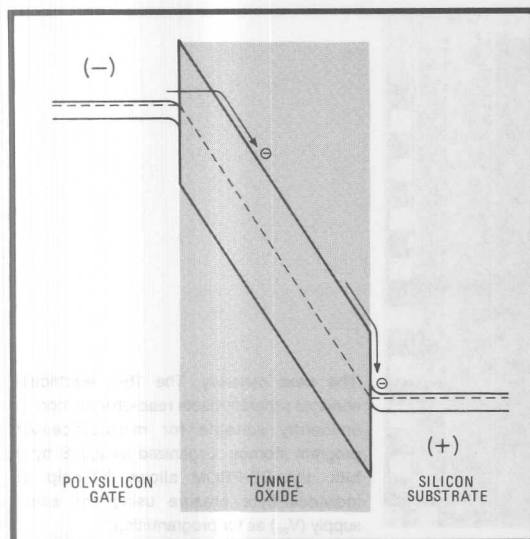
10^{-21} A or less must be guaranteed during read or storage operations. The ratio of these currents, $1:10^{11}$, represents a difficult design problem. Few charge-injecting mechanisms are known that can be turned off reliably during nonprogram periods for such a ratio.

One structure that has proven capable of meeting such stringent reliability requirements has done so for many millions of devices over the last nine years. This is the floating-gate avalanche-injection MOS (Famos) device used in the 1702, 2708, 2716, and 2732 E-PROM families. In the Famos structure, shown in Fig. 1a, a polysilicon gate is completely surrounded by silicon dioxide, one of the best insulators around. This ensures the low leakage and long-term data retention.

To charge the floating gate, electrons in the underlying MOS device are excited by high electric fields in the channel, enabling them to jump the silicon/silicon-dioxide energy barrier between the substrate and the thin gate dielectric. Once they penetrate the gate oxide, the electrons flow easily toward the floating gate as it was previously capacitively coupled with a positive bias to attract them.

Because of Famos' proven reliability, the floating-gate approach was favored for the EE-PROM. The problem, of course, was to find a way to discharge the floating gate electrically. In an E-PROM, this discharge is effected by exposing the device to ultraviolet light. Electrons absorb photons from the UV radiation and gain enough energy to jump the silicon/silicon-dioxide energy barrier in the reverse direction as they return to the substrate. This suffices for off-board program rewriting, but the object of the EE-PROM is to satisfy new applications that demand numerous alterations of the stored data without removing the memory from its system environment. What evolved was the new cell structure called Flotox (Fig. 1b).

In the quest for electrical erasability, many methods were considered, and several potentially viable solutions were pursued experimentally. One initially attractive



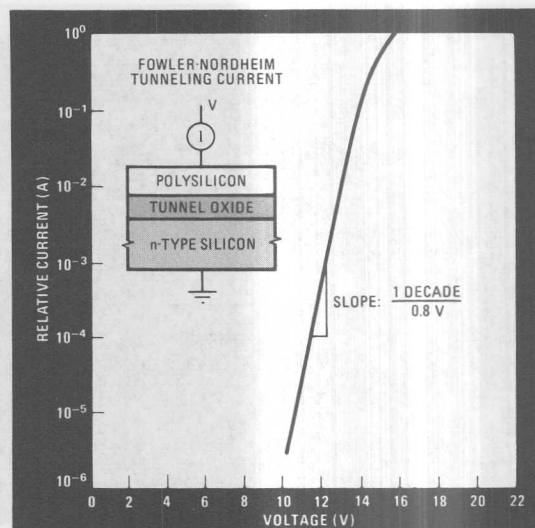
2. Tunneling. For a thin enough oxide, as shown here, under a field strength of 10^7 V/cm, Fowler-Nordheim tunneling predicts that a certain number of electrons will acquire enough energy to jump the forbidden gap and make it from the gate to the substrate.

approach attempts to harness a parasitic charge-loss mechanism discovered in the earliest E-PROMs. Referring again to Fig. 1a, the polysilicon grains on the top surface of the floating gate tend, under certain processing conditions, to form sharp points called asperities. The sharpness of the asperities creates a very high local electric field between the polysilicon layers, shoving electrons from the floating gate toward the second level of polysilicon. This effect is purposely subdued in today's E-PROMs by controlling oxide growth on top of the floating gate because this parasitic electron-injection mechanism would otherwise interfere with proper E-PROM programming.

It was first thought that asperity injection could be used to erase the chip. In fact, fully functional, electrically erasable test devices were produced; but the phenomenon proved unreproducible and the devices tended to wear out quickly after repeated program and erase cycling. After over a year's effort, that approach was abandoned.

Tunneling solution

The solution turned out to be the one that initially seemed impossible. After investigating many methods of producing energetic electrons, it was decided to approach the problem from a different direction: to pass low-energy electrons through the oxide. This could be accomplished through Fowler-Nordheim tunneling, a well-known mechanism, depicted by the band diagram in Fig. 2. Basically, when the electric field applied across an insulator exceeds approximately 10^7 volts per centimeter, electrons from the negative electrode (the polysilicon in Fig. 2) can pass a short distance through the forbidden gap of the insulator and enter the conduction band. Upon their arrival there, the electrons



3. Current characteristic. In Fowler-Nordheim tunneling, current flow depends strongly on voltage across the oxide, rising an order of magnitude for every 0.8 V. Charge retention is adequate so long as the difference between programming and reading is at least 8.8 V.

flow freely toward the positive electrode.

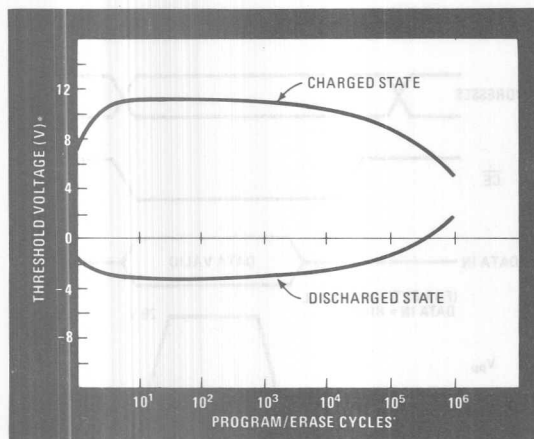
This posed two fundamental problems. First, it was commonly believed that silicon dioxide breaks down catastrophically at about 10^7 V/cm, and MOS FETs are normally operated at field strengths 10 times below this. Second, to induce Fowler-Nordheim tunneling at reasonable voltages (20 V), the oxide must be less than 200 angstroms thick. Oxide thickness below about 500 Å had rarely even been attempted experimentally, and it was feared that defect densities might prove prohibitively high.

To be weighed against these risks, however, were several advantages. Tunneling in general is a low-energy, efficient process that eliminates power dissipation. Fowler-Nordheim tunneling in particular is bilateral and can be used for charging the gate as well as discharging it. Finally, the tunnel oxide area could be made very small, which is of course consistent with the needs of high-density processing.

With these motivating factors, development was initiated to grow reliable, low-defect oxides less than 200 Å thick. The success of this effort resulted in the realization of a working cell structure called Flotox.

The Flotox device cross section is pictured in Fig. 1b. It resembles the Famos structure except for the additional tunnel-oxide region over the drain. With a voltage V_g applied to the top gate and with the drain voltage V_d at 0 V, the floating gate is capacitively coupled to a positive potential. Electrons are attracted through the tunnel oxide to charge the floating gate. On the other hand, applying a positive potential to the drain and grounding the gate reverses the process to discharge the floating gate.

Flotox, then, provides a simple, reproducible means for both programming and erasing a memory cell. But



4. Good endurance. The endurance of the EE-PROM depends on the threshold-voltage difference between the charged and discharged states. Though repeated cycling degrades thresholds, the chip should stay within tolerable limits for 10⁴ to 10⁶ cycles.

what about charge retention and refresh considerations with such a thin oxide? The key to avoiding such problems is given in Fig. 3, which shows the exceedingly strong dependence of the tunnel current on the voltage across the oxide. This is characteristic of Fowler-Nordheim tunneling.

The current in Fig. 3 rises one order of magnitude for every 0.8-V change in applied voltage. If the 11 orders of magnitude requirement is recalled, it is apparent that the difference between the voltage across the tunnel oxide during programming and that during read or storage operations must be in excess of 8.8 V.

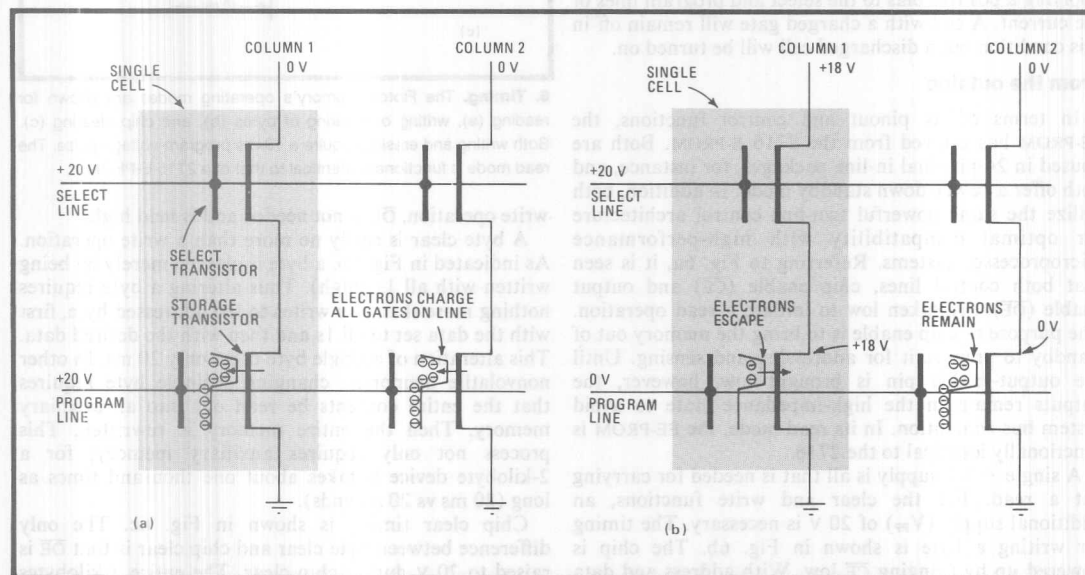
This value, including margins for processing variations, is reasonable. Furthermore, data is not disrupted during reading or storage so that no refreshing is required under normal operating or storage conditions. Extensive experimental testing has verified that data retention exceeding 10 years at a temperature of 125°C is possible.

Another important consideration is the behavior of the electrically erasable memory cell under repeated program erase cycling. This is commonly referred to as endurance. The threshold voltage of a typical Flotex cell, in both the charged and discharged states, is shown in Fig. 4 as a function of the number of programming or erasing cycles. There is some variation in the threshold voltages with repeated cycling but this remains within tolerable limits out to very high numbers of cycles—somewhere between 10⁴ and 10⁶ cycles.

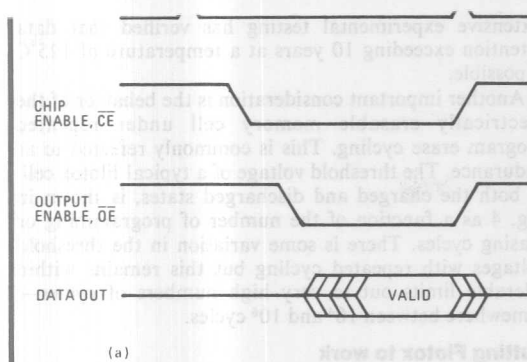
Putting Flotex to work

The Flotex cell is assembled into a memory array using two transistors per cell as shown in Fig. 5. The Flotex device is the actual storage device, whereas the upper device, called the select transistor, serves two purposes. First, when discharged, the Flotex device exhibits a negative threshold. Without the select transistor, this could result in sneak paths for current flow through nonselected memory cells. Secondly, the select transistor prevents Flotex devices on nonselected rows from discharging when a column is raised high.

The array must be cleared before information is entered. This returns all cells to a charged state as shown schematically in Fig. 5a. To clear the memory all the select lines and program lines are raised to 20 V while all the columns are grounded. This forces electrons through the tunnel oxide to charge the floating gates on all of the



5. Working. To clear a Flotex cell, select and program lines are raised to 20 V and columns are grounded (a). To write a byte of data, the program line is grounded and the columns of the selected byte are raised or lowered according to the data pattern (b).



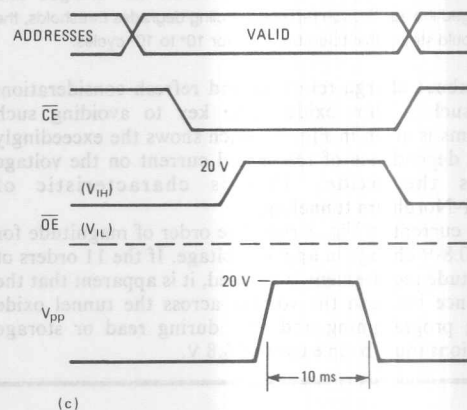
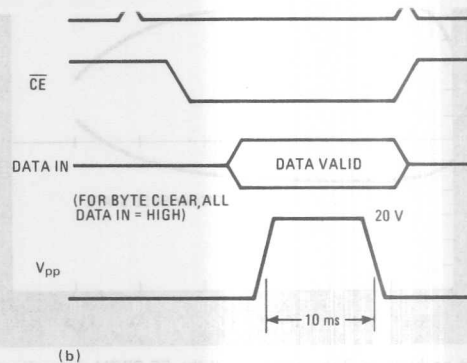
selected rows. An advantage of this EE-PROM over E-PROMs is the availability of both byte- and chip-clear operations. The byte-clear one is particularly useful for a memory of this size. When it is initiated, only the select and program lines of an addressed byte rise to 20 V.

To write a byte of data, the select line for the addressed byte is raised to 20 V while the program line is grounded as shown in Fig. 5b. Simultaneously, the columns of the selected byte are raised or lowered according to the incoming data pattern. The bit on the left in Fig. 5b, for example, has its column at a high voltage, causing the cell to discharge, whereas the bit on the right has its column at ground so its cell will experience no change. Reading is accomplished by applying a positive bias to the select and program lines of the current. A cell with a charged gate will remain off in this condition but a discharged cell will be turned on.

From the outside

In terms of its pinout and control functions, the EE-PROM has evolved from the 2716 E-PROM. Both are housed in 24-pin dual in-line packages, for instance, and both offer a power-down standby mode. In addition, both utilize the same powerful two-line control architecture for optimal compatibility with high-performance microprocessor systems. Referring to Fig. 6a, it is seen that both control lines, chip enable (\overline{CE}) and output enable (\overline{OE}), are taken low to initiate a read operation. The purpose of chip enable is to bring the memory out of standby to prepare it for addressing and sensing. Until the output-enable pin is brought low, however, the outputs remain in the high-impedance state to avoid system bus contention. In its read mode, the EE-PROM is functionally identical to the 2716.

A single +5-v supply is all that is needed for carrying out a read. For the clear and write functions, an additional supply (V_{PP}) of 20 V is necessary. The timing for writing a byte is shown in Fig. 6b. The chip is powered up by bringing \overline{CE} low. With address and data applied, the write operation is initiated with a single 10-ms, 20-V pulse applied to the V_{PP} pin. During the



6. Timing. The Flotex memory's operating modes are shown for reading (a), writing or clearing of bytes (b), and chip clearing (c). Both writing and erasing require a 10-ms program-voltage pulse. The read mode is functionally identical to that of a 2716 E-PROM.

write operation, \overline{OE} is not needed and is held high.

A byte clear is really no more than a write operation. As indicated in Fig. 6b, a byte is cleared merely by being written with all 1s (high). Thus altering a byte requires nothing more than two writes to the addressed byte, first with the data set to all 1s and then with the desired data. This alteration of a single byte takes only 20 ms. In other nonvolatile memories, changing a single byte requires that the entire contents be read out into an auxiliary memory. Then the entire memory is rewritten. This process not only requires auxiliary memory; for a 2-kilobyte device it takes about one thousand times as long (20 ms vs 20 seconds).

Chip clear timing is shown in Fig. 6c. The only difference between byte clear and chip clear is that \overline{OE} is raised to 20 V during chip clear. The entire 2 kilobytes are cleared with a single 10-ms pulse. Addresses and data are not all involved in a chip-clear operation. □

SESSION XII: ROMs, PROMs AND EROMs

THPM 12.4: A 35ns 16K PROM

Robert K. Wallace, Arthur J. Learn and Kenneth W. Schuette

Intel Corp.

Santa Clara, CA

THE UTILIZATION of positive resist photolithography in conjunction with polysilicon fuses and two level metalization has resulted in performance improvement and die size reduction in bipolar PROMs. A 16,384b PROM organized 2K x 8 has been designed and fabricated on a 140mil square chip which has 25ns typical access time and 600mW power dissipation.

A standard diffused isolation Schottky bipolar technology was used as the basis for this technology because of a long history in manufacturing. Positive resist projection lithography was used to give 3μ features on key layers such as base, emitter and contacts. The polysilicon layer used to form fuses is also used over emitter regions both to self-align the contact and emitter and to provide protection against junction spiking which can occur in self-aligned structures. A cross section of the basic transistor structure is shown in Figure 1.

A base-emitter diode was chosen for the memory cell because it takes maximum advantage of the polysilicon fuse material by making a direct contact to the emitter region. The base-emitter diode in an emitter follower array also has the advantage of having current gain, relatively good conductance per square μ of area and of being self-isolated. Two level metalization is utilized to enhance the density and to deliver uniform programming current to the very large memory array. The first level metal is used as word lines, while second level metal serves as bit lines. The second level metalization makes a direct contact to the fuse for each bit to further enhance programming current uniformity. A photomicrograph of the memory chip is shown in Figure 2.

The fabrication of large dense memory arrays places special constraints on circuit design in two key areas. First, a decoder must be designed which meets speed, power and memory cell pitch requirements. This is accomplished by using the decoder shown in Figure 3. Schottky diodes are utilized to minimize the decoder's internal node capacitance and to provide a relatively low impedance logic low level. A lateral PNP is incorporated into the decoder to increase the drive capability of the decoder during programming. Second, a current sense amplifier is needed which requires minimal voltage swing at the very heavily loaded sense node. This has been accomplished by the circuit shown in Figure 4 in which a current source and common base stage have been employed to minimize the required logic swing to less than 500mV. An oscillograph of the address to output delay is shown in Figure 5.

Acknowledgments

The authors would like to thank L. Calhoun, L. Furtado, P. Poenisch, P. Schoen and G. Serhan for their important contributions to this project.

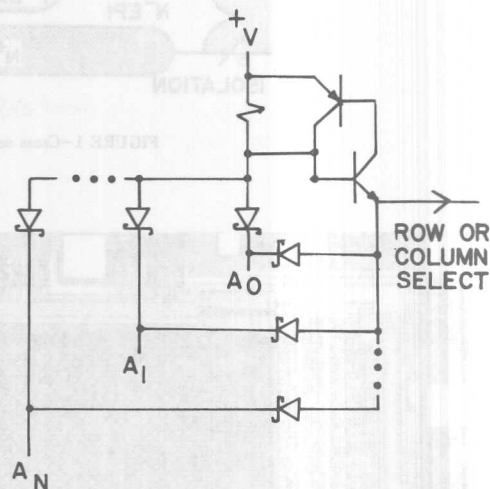


FIGURE 3—Schematic diagram of row or column decoder.

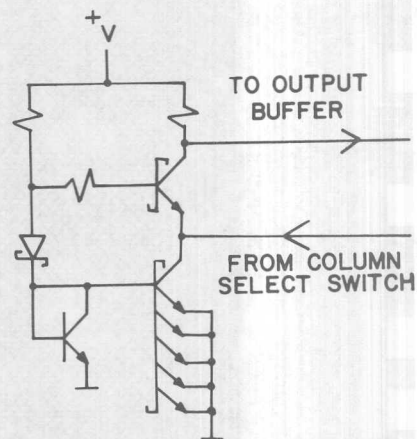


FIGURE 4—Schematic diagram of sense amplifier.

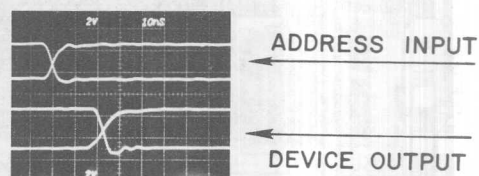


FIGURE 5—Oscillograph showing typical device address to output delay. Operating conditions are $T = 23^\circ\text{C}$ and $V_{cc} = 5.0\text{V}$.

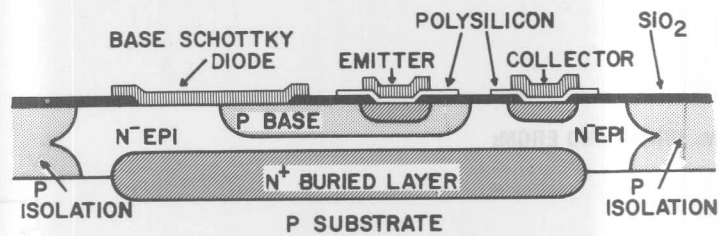


FIGURE 1—Cross section showing basic transistor structure.

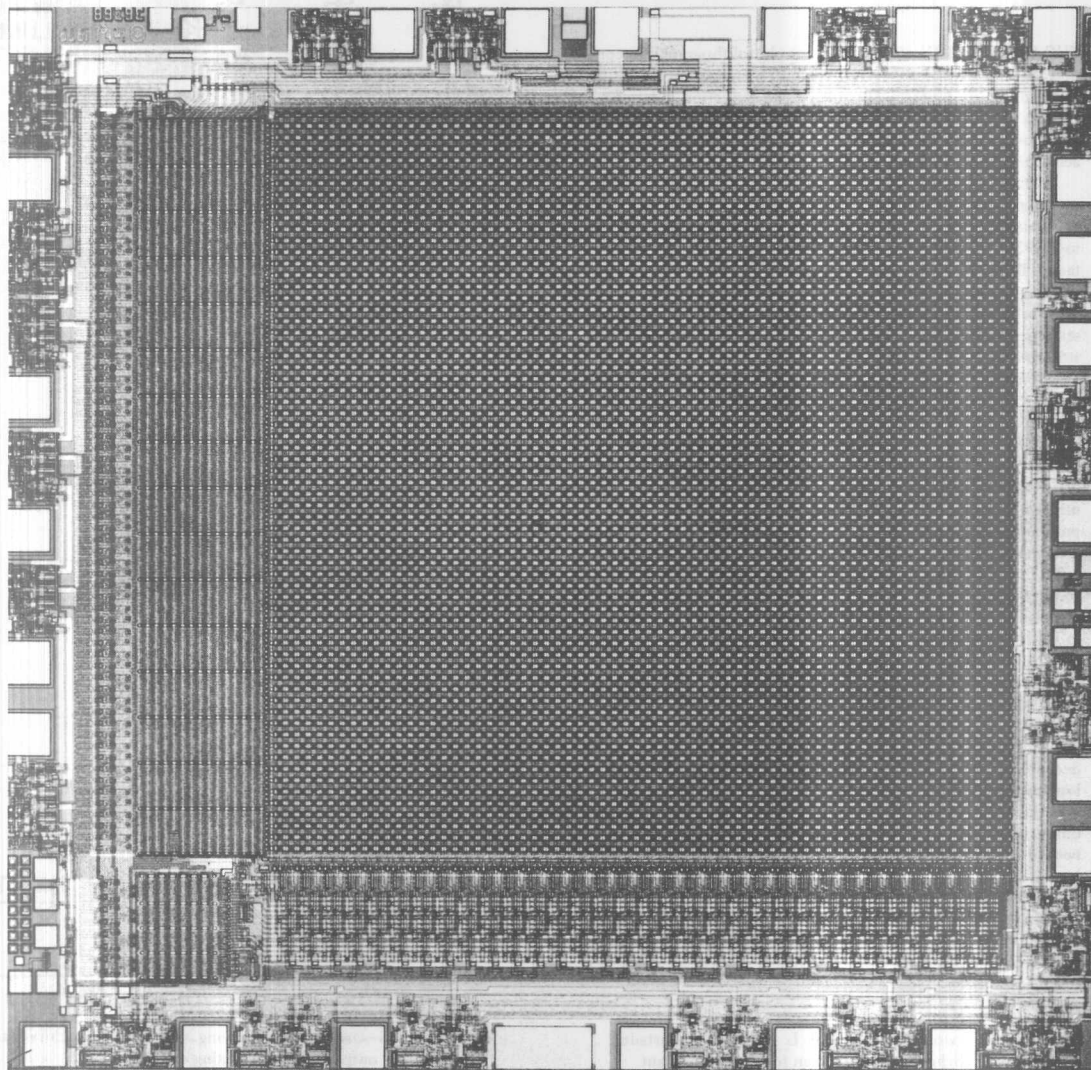


FIGURE 2—Photomicrograph of the 16K PROM.